promised, it should be held responsible. Of course, we must not be too rigid about this separation, since it is clearly desirable to encourage manufacturing to be flexible enough to accommodate legitimate changes from sales. Chapter 15 will probe this problem in greater detail and will give specifics on how to quote customer due dates sensibly and derive a set of manufacturing due dates from them.

The disparity between responsibility and authority can extend beyond the workers into management and can be the result of subtle factors. We witnessed an example of a particular manager who had responsibility for the operational aspects of his production line, including throughput, quality, and cycle time. Moreover, he had full authority, budgetary and otherwise, to take the necessary steps to achieve his performance targets. However, he was unable to do so because of a lack of *time* to spend on operational issues; he was also responsible for personnel issues for the workforce on the line, and the majority of his time was taken up with these concerns. As a result, he was taking a great deal of heat for the poor operational performance of his line. Our impression is that this is not at all an unusual situation.

To avoid placing managers in a position in which they are unable to deal effectively with logistical concerns, we suggest using policies to explicitly *make time for operations*. One approach is to designate a manager as the "operating manager" for a specific period (e.g., a shift or day). During this time, the manager is temporarily exempted from personnel duties and is expected to concentrate exclusively on running the line. The effect will be to force the manager to appreciate the problems at an intimate level and provide time for generating solutions. This concept is analogous to the "officer of the deck" (OOD) policy used in navies around the world. When the OOD "has the con," he is ultimately responsible for the operation of the ship and is temporarily absolved from all duties not directly related to this responsibility. On a ship, having a clearly defined ultimate authority at all times is essential to making critical decisions on a split-second basis. As manufacturing practice moves toward low-WIP, short-cycle-time techniques, having a manager with the time and focus to make real-time judgments on operating issues will become increasingly important in factories as well.

11.5 Summary

We realize that this chapter is only a quick glance at the complex and multifaceted manner in which human beings function in manufacturing systems. We hope we have offered enough to convince the reader that operations management is more than just models. Even strongly technical topics, such as scheduling, capacity planning, quality control, and machine maintenance, involve people in a fundamental way. It is important to remember that a manufacturing system consists of equipment, logic, *and* people. Well-designed systems make effective use of all three components.

Beyond this fundamental observation, our main points in this chapter were these:

1. *People act according to their self-interest.* Certainly altruism exists and sometimes motives are subtle, but overwhelmingly, peoples' actions are a consequence of their real and perceived personal incentives. If these incentives induce behavior that is counterproductive to the system, they must be changed. While we cannot give here any kind of comprehensive treatment of the topic of motivation, we have tried to demonstrate that simple financial incentive systems are unlikely to be sufficient.

2. *People differ.* Because individuals differ with regard to their talents, interests, and desires, different systems are likely to work with different workforces. It makes no

sense to force-fit a control system to an environment in which the workers' abilities are ill suited to it.

3. *Champions can have powerful positive and negative influences.* We seem to be in an age when each new manufacturing management idea must be supported by a guru of godlike stature. While such people can be powerful agents for change, they can also make unsound ideas seem attractive. We would all probably be better off with a little less hype and a little more plodding, incremental improvement in manufacturing.

4. *People can burn out.* This is a real problem for the post-1990 era. We have jumped on so many bandwagons that workers and managers alike are tired of the "revolution of the month." In the future, promoting real change in manufacturing plants is likely to require less reliance on rhetoric and more on logic and hard work.

5. *There is a difference between planning and motivating.* Using optimistic capacity, yield, or reliability data for motivational purposes may be appropriate, provided it is not carried to extremes. But using historically unproven numbers for predictive purposes is downright dangerous.

6. *Responsibility should be commensurate with authority.* This well-known and obvious management principle is still frequently violated in manufacturing practice. In particular, as we move toward more rapid, low-WIP manufacturing styles, it will be increasingly important to provide managers with *time* for operations as part of their authority for meeting their manufacturing responsibilities.

We hope that these simple observations will inspire the reader to think more carefully about the human element in operations management systems. We have tried to maintain a human perspective in Part III of this book, in which we discuss putting the factory physics concepts into practice, and we encourage the reader to do the same.

Discussion Points

1. Comment on the following paraphrase of a statement by an hourly worker overheard in a plant lunchroom:

> Management expects us to bust our butts getting more efficient and reengineering the plant. If we don't, they'll be all over us. But if we do, we'll just downsize ourselves out of jobs. So the best thing to do is make it look like we're working real hard at it, but be sure that no really big changes happen.

a. What does this statement imply about the relationship between management and labor at that plant?
b. Does the worker have a point?
c. How might such concerns on the part of workers be addressed as part of a program of change?

2. Consider the following paraphrase of a statement by the owner of a small manufacturing business:

> Twenty years ago our machinists were craftsmen and knew these processes inside and out. Today, we're lucky if they show up on a regular basis. We need to develop an automated system to control the process settings on our machines, not so much to enhance quality or keep up with the competition, but because the workers are no longer capable of doing it manually.

a. What does this statement imply about the relationship between management and labor at that plant?

b. Does the owner have a point?
c. What kinds of policies might management pursue to improve the effectiveness of operators?

3. Consider the following statement:

> JIT worked for Toyota and other Japanese companies because they had the champions who originated it. American firms were far less successful with it because they had less effective champions to sell the change.

a. Do you think there is a grain of truth in this statement?
b. What important differences about JIT in Japan and America does it ignore?

Study Questions

1. The popular literature on manufacturing has sometimes portrayed continual improvement as a matter of "removing constraints." Why are constraints sometimes a good thing in manufacturing systems? How could removing constraints actually make things worse?
2. When dealing with a manufacturing system that is burned out by "revolutions," what measures can a manager use to inspire needed change?
3. Many manufacturing managers are reluctant to use historical capacity data for future planning because they regard it as tantamount to accepting previous substandard performance. Comment on the dilemma between using historical capacity data for planning versus using rated capacity for motivation. What measures can a manager take to separate planning from motivation?
4. In Deming's red beads example, employees have no control over their performance. What does this experiment have to do with a situation in the real world, where employees' performance is a function both of their ability/effort and random factors? What managerial insights can one obtain from this example?
5. Contrast MRP, Kanban, and CONWIP from a human issues standpoint. What implications do each of these systems have for the working environment of the employees on the factory floor? The staff engineers responsible for generating and propagating the schedule? The managers responsible for supervising direct labor? To what extent are the human factors benefits of a particular production control system specific to that system, and therefore not to be obtained by modifying one of the other production control methods?

CHAPTER

12 TOTAL QUALITY MANUFACTURING

Saw it on the tube
Bought it on the phone
Now you're home alone
It's a piece of crap.

I tried to plug it in
I tried to turn it on
When I got it home
It was a piece of crap.
Neil Young

12.1 Introduction

A fundamental factory physics insight is that variability plays an important role in determining the performance of a manufacturing system. As we observed in Chapters 8 and 9, variability can come from a variety of sources: machine failures, setups, operator behavior, fluctuations in product mix, and many others. A particularly important source of variability, which can radically alter the performance of a system, is quality. Quality problems almost always become variability problems. By the same token, variability reduction is frequently a vehicle for quality improvement. Since quality and variability are intimately linked, we conclude Part II with an overview of this critical issue.[1]

12.1.1 The Decade of Quality

The 1980s were the *decade of quality* in America. Scores of books were published on the subject, thousands of employees went through short courses and other training programs, and "quality-speak" entered the standard language of corporate America. In 1987, the International Standards Organization established the ISO 9000 Series of quality

[1] We have deliberately used the title *Total Quality Manufacturing* in place of the more conventional *Total Quality Management (TQM)* in recognition that we are covering only the subset of TQM that relates to operations management.

standards. In the same year, the Malcolm Baldrige National Quality Award was created by an act of the U.S. Congress.[2]

The concept of quality and the methods for its control, assurance, and management were not new in the 1980s. Quality control as a discipline dates back at least to 1924 when Walter A. Shewhart of Western Electric's Bell Telephone Laboratories first introduced process control charts. Shewhart published the first important text on quality in 1931. Armand Feigenbaum coined the term *total quality control* in a 1956 paper and used this as the title of a 1961 revision of his 1951 book, *Quality Control.*

But while the terms and tools of quality have been around for a long time, it was not until the 1980s that American industry really took notice of the strategic potential of quality. Undoubtedly, this interest was stimulated in large part by the dramatic increase in the quality of Japanese products during the 1970s and 1980s, much in the same way that American interest in inventory reduction was prompted by Japanese JIT success stories.

Has all the talk about quality led to improvements? Probably so, although it is difficult to measure them since, as we will discuss in this chapter, quality is a broad term that can be interpreted in many ways. Nevertheless, some surveys have suggested that consumers viewed the overall quality of American products as *declining* during the 1980s (Garvin 1988). The American Customer Satisfaction Index (ACSI), an overall gauge of customer perceptions of quality that has been tracked quarterly since 1994, also showed declining satisfaction in the 1990s. Whether these declines are due to rising customer expectations, ongoing management problems, or both, it seems clear that quality remains a significant challenge for the future.

12.1.2 A Quality Anecdote

To set the stage, we introduce the quality issue from a personal perspective. In 1991, one of the authors purchased a kitchen range that managed to present an astonishing array of quality problems. First, for styling purposes, the stove came with light-colored porcelain-coated steel cooktop grates. After only a few days of use, the porcelain cracked and chipped off, leaving a rough, unattractive appearance. When the author called the service department (and friends with similar stoves), he found that *every single* stove of this model suffered from the same defect—a 100 percent failure rate! So much for inspection and quality assurance!

The customer service department was reasonably polite and sent replacement grates, but these lasted no longer than the originals, so the author continued to complain. After three or four replacements (including one in which the service department sent two sets with the recommendation that we use one set and save the other to put on the stove when entertaining guests!), the manufacturer changed suppliers and sent dark-colored, more durable grates. So much for quality design and styling!

As the grate story was evolving, the stove suffered from a succession of other problems. For instance, the pilotless ignition feature would not shut off after the burners lit, causing a loud clicking noise whenever the stove was in use. Repair people came to fix this problem no fewer than *eight* times during the first year of use (i.e., the warranty period). During one of these visits, the repairman admitted that he really had no idea of how to adjust the stove because he had never received specifications for this model from the manufacturer and was therefore just replacing parts and hoping for the best. So much for service after the sale and for doing things right the first time!

[2] Tellingly, the Japanese Union of Scientists and Engineers (JUSE) had already established its major quality award, the Deming Prize, in honor of American W. Edwards Deming, in 1951.

At the end of the first year, the service department called to sell the author an extended warranty and actually said that because the stove was so unreliable (they used a much less polite term than *unreliable*) the extended warranty would be a good deal for us. So much for standing behind your product, and for customer-driven quality!

(By the way, as this book was being written, the oven door fell off. WE DID NOT MAKE ANY OF THIS UP!)

12.1.3 The Status of Quality

We do not mean to imply that this story sums up the quality level of American manufacturing. But it is fascinating (and depressing) that a company in the 1990s could be in such glaring violation of virtually every principle of good quality management. Furthermore, we suspect that this is not an isolated example. (We have more personal experiences, but will not subject the reader to them.) An exercise we are fond of in our executive courses is to challenge the participants to think about quality not from the viewpoint of a manufacturing manager, educator, or professional, but from that of a *consumer*. A dishearteningly high fraction report that they have rarely had their expectations for a product or service exceeded, but have frequently been disappointed.

Evidently, there is still a considerable gap between the rhetoric and the reality of quality. Thus, while it is convenient to speak, as we did at the beginning of the book, as if *cost* was the dimension of competition in the 1970s, *quality* was the dimension of competition in the 1980s, and *speed* is the dimension of competition in the 1990s, one should not take this apothegm literally. Quality (and cost, for that matter) will remain an important determinant of competitiveness well beyond the 1990s.

What can an individual firm do? The answer is, plenty. There is not a plant in the world that could not improve its products, processes, or systems; get closer to its customers; or better understand the influence of quality on its business. Furthermore, there is a vast literature to consult for ideas. Although the quality literature, like the JIT literature, contains an overabundance of imprecise romantic rhetoric, it offers much useful guidance as well. The literature on quality can be divided into two categories, **total quality management (TQM),** which focuses on quality in qualitative management terms (e.g., fostering an overall environment supportive of quality improvement), and **statistical quality control (SQC),** which focuses on quality in quantitative engineering terms (e.g., measuring quality and assuring compliance with specifications). Both views are needed to formulate an effective quality improvement program. All TQM with no SQC produces talk without substance, while all SQC with no TQM produces numbers without purpose.

A strong representative from the TQM literature is the work of Garvin (1988), on which some of the following discussion is based. Garvin's book offers an insightful perspective of what quality is and how it affects the firm. Other widely read TQM books include those by Crosby (1979, 1984), Deming (1986), and Juran (1989, 1992). In the SQC field there are many solid works, most of which contain a brief introductory section on TQM; these include those by Banks (1989); DeVor, Chang, and Sutherland (1992); Gitlow et al. (1989); Montgomery (1991); and Thompson and Koronacki (1993); among others. Some books, notably Juran's *Quality Control Handbook* (1988), address both the TQM and SQC perspectives.

We cannot hope to provide the depth and breadth of these references in this brief chapter. What we can do is to focus on how quality fits into the overall picture of plant operations management. The framework of factory physics allows us to synthesize the perspectives of quality and operations into elements of the same picture. We leave the reader to consult references like those mentioned, to flesh out the specifics of quality management procedures.

12.2 Views of Quality

12.2.1 General Definitions

What is quality? This question is a logical place to start our discussion. Garvin (1988) offers five definitions of quality, which we summarize as follows:

1. *Transcendent.* Quality refers to an "innate excellence," which is not a specific attribute of either the product or the customer, but is a third entity altogether. This boils down to the "I can't define it, but I know it when I see it" view of quality.
2. *Product-based.* Quality is a function of the attributes of the product (the quality of a rug is determined by the number of knots per square inch, or the quality of an automobile bumper is determined by the dollars of damage caused by a five-mile-per hour crash). This is something of a "more is better" view of quality (more knots, more crashworthiness, etc.).
3. *User-based.* Quality is determined by how well customer preferences are satisfied; thus, it is a function of whatever the customer values (features, durability, aesthetic appeal, and so on). In essence, this is the "beauty is in the eye of the beholder" view of quality.
4. *Manufacturing-based.* Quality is equated with conformance to specifications (e.g., is within dimensional tolerances, or achieves stated performance standards). Because this definition of quality directly refers to the processes for making products, it is closely related to the "do it right the first time" view of quality.
5. *Value-based.* Quality is jointly determined by the performance or conformance of the product *and* the price (e.g., a $1,000 compact disk is not high quality, regardless of performance, because few would find it worth the price). This is a "getting your money's worth" or "affordable excellence" view of quality.

These definitions bring up two points. First, quality is a multifaceted concept that does not easily reduce to simple numerical measures. We need a framework within which to evaluate quality policies, just as we needed one (i.e., factory physics) for evaluating operations management policies. Indeed, as we will discuss, the two frameworks are closely related, perhaps as two facets of the larger science of manufacturing to which we referred in Chapter 6.

Second, the definitions are heavily **product-oriented.** This is the case with most of the TQM literature and is a function of the principle that quality must ultimately be "customer-driven." Since what the customer sees is the product, quality must be measured in product terms. However, the quality of the product as seen by the customer is ultimately determined by a number of **process-oriented** factors, such as design of the product, control of the manufacturing operations, involvement of labor and management in overseeing the process, customer service after the sale, and so on.

12.2.2 Internal versus External Quality

To better understand the relationship between product-oriented and process-oriented quality, we find it useful to draw the following distinction between internal quality and external quality:

1. **Internal quality** refers to conformance with quality specifications inside the plant and is closely related to the manufacturing-based definition of quality. It is typically monitored through direct product measures such as scrap and rework rates and indirect process measures such as pressure (in an injection molding machine) and temperature (in a plating bath).
2. **External quality** refers to how the customer views the product and may be interpreted by using the transcendent, product-based, user-based, or value-based definition, or a combination of them. It can be monitored via direct measures of customer satisfaction, such as return rate, and indirect indications of customer satisfaction derived from sampling, inspection, field service data, customer surveys, and so on.

To achieve high external quality, one must translate customer concerns to measures and controls for internal quality. Thus, from the perspective of a manufacturing manager, the links between internal and external quality are key to the development of a strategically effective quality program. The following are some of the more important ways in which quality inside the plant is linked to the quality that results in customer satisfaction.

1. *Error prevention.* If fewer errors are made in the plant, fewer defects are likely to slip through the inspection process and reach the customer. Therefore, to the extent that quality as perceived by the customer is determined by freedom from defects, high "quality at the source" in the plant will engender high customer-driven quality.

2. *Inspection improvement.* If fewer defects are produced during the manufacturing process, then quality assurance will require inspection to detect and reject or correct fewer items. This tends to reduce pressure on quality personnel to "let things slide"—in other words, relax quality standards in the name of getting product out the door.[3] Furthermore, the less time spent reworking or replacing defective parts, the more time people have for tracing quality problems to the root causes. Ideally, the net effect will be an upward quality spiral, in which error prevention and error detection both improve over time.

3. *Environment enhancement.* Even if quality problems in the field cannot be traced directly to plant-level defects, high internal quality and external quality may still be linked.[4] Both types of quality are promoted by the same environmental factors (e.g., supportive management attitudes, tangible rewards for improvements, sophisticated tracking and control systems, and effective training). An organization that has fostered the right attitudes and tools inside the plant is likely to be able to do the same outside the plant.

In short, understanding quality means looking to the customer. Delivering it entails looking to manufacturing.[5] For the purposes of this chapter we will assume that the concerns of the customer have been understood and translated to quality specifications for use by the plant. Our focus will be on the relationship between quality and operations, and particularly how the two can work together as parts of a continual improvement process for the plant.

[3] Crosby (1979, 41) relates a story in which manufacturing viewed inspection in an adversarial mode, protesting each rejected part as if quality inspectors were personally trying to sabotage the plant.

[4] Garvin (1988, 129) offers the example of the compressor on an air conditioner failing due to corrosion caused by excess moisture seeping into the unit. Such a problem would not show up in any reasonable "burn in" period and therefore would most likely be undetected as a defect at the plant level.

[5] Here we are referring to "big M" manufacturing, including product design, production, and field service.

12.3 Statistical Quality Control

Statistical quality control (SQC) generally focuses on *manufacturing* quality, as measured by conformance to specifications. The ultimate objective of SQC is the systematic reduction of variability in key quality measures. For instance, size, weight, smoothness, strength, color, and speed (e.g., of delivery) are all measurable attributes that can be used to characterize the quality of manufacturing processes. By working to assure that these measures are tightly controlled within desired bounds, SQC functions directly at the interface between operations and quality.

12.3.1 SQC Approaches

There are three major classes of tools used in SQC to ensure quality:

1. *Acceptance sampling.* Products are inspected to determine whether they conform to quality specifications. In some situations, 100 percent inspection is used, while in others some form of statistical sampling is substituted. Sampling may be an option chosen for cost reasons or an absolute necessity (e.g., when inspection is destructive).
2. *Process control.* Processes are continuously monitored with respect to both mean and variability of performance to determine when special problems occur or when the process has gone out of control.
3. *Design of experiments.* Causes of quality problems are traced through specifically targeted experiments. The basic idea is to systematically vary controllable variables to determine their effect on quality measures. A host of statistical tools (e.g., block designs, factorial designs, nested designs, response surface analysis, and Taguchi methods) have been developed for efficiently correlating controls with outputs and optimizing processes.

Typically, as an organization matures, it relies less on after-the-fact acceptance sampling and more on at-the-source process control and continual-improvement-oriented design of experiments.

Obviously, entire books have been written on each of these subjects, so detailed coverage of them is beyond the scope of this chapter. However, because process control deals so specifically with the interface between quality and variability, we offer an overview of the basic concepts here.

12.3.2 Statistical Process Control

Statistical process control (SPC) begins with a measurable quality attribute—for example, the diameter of a hole in a cast steel part. Regardless of how tightly controlled the casting process is, there will always be a certain amount of variability in this diameter. If it is relatively small and due to essentially uncontrollable sources, then we call it **natural variability.** A process that is operating stably within its natural variation is said to be **in statistical control.** Larger sources of variability that can potentially be traced to their causes are called **assignable-cause variation.** A process subject to assignable-cause variation is said to be **out of control.** The fundamental challenge of SPC is to separate assignable-cause variation from natural variation. Because we generally observe directly only the quality attribute itself, but not the causes of variation, we need statistics to accomplish this.

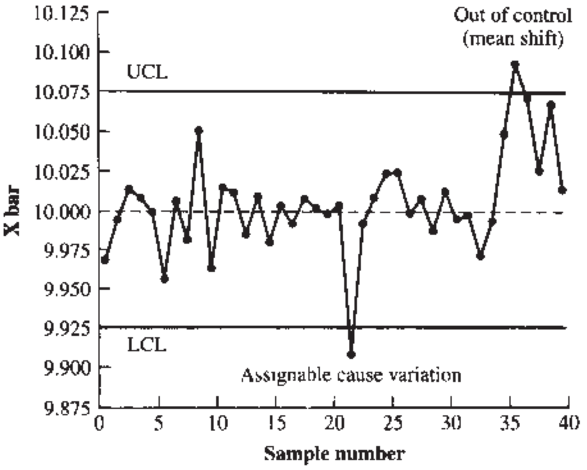


Figure 12.1
Process control chart for average hole size in steel castings

To illustrate the basic principles behind SPC, let us consider the example of controlling the diameter of a hole in a steel part made using a sand casting process. Suppose that the desired nominal diameter is 10 millimeters and we observe a casting with a diameter of 10.1 millimeters. Can we conclude that the casting process is out of control? The answer is, of course, "It depends." It may be that a deviation of 0.1 millimeter is well within natural variation levels. If this were the case and we were to adjust the process (e.g., by altering the sand, steel, or mold) in an attempt to correct the deviation, in all likelihood we would make it worse. The reason is that adjusting a process in response to random noise increases its variability (see Deming 1982, 327, for discussion of a funnel experiment that illustrates this point). Hence, to ensure that adjustments are made only in response to assignable-cause variation, we must characterize the natural variation.

In our example, suppose we have measured a number of castings and have determined that the mean diameter can be controlled to be $\mu = 10$ millimeters and the standard deviation of the diameter is $\sigma = 0.025$ millimeter. Further suppose that every two hours we take a random sample of five castings, measure their hole diameters, compute the average (which we call $\bar{x}$), and plot it on a chart like that shown in Figure 12.1. From basic statistics, we know that $\bar{x}$ is itself a random variable which has standard deviation

$$\sigma_{\bar{x}} = \frac{\sigma}{\sqrt{n}} \tag{12.1}$$

where n is the number in the sample; $n = 5$ in this example.[6]

The basic idea behind control charts is very similar to hypothesis testing. Our null hypothesis is that the process is in control; that is, the samples are coming from a process with mean μ and standard deviation σ. To avoid concluding that the process is out of control when it is not (i.e., type I error), we set a stringent standard for designating deviations as "assignable cause." Standard convention is to flag points that lie more than three standard deviations above or below the mean. We do this by specifying lower and upper control limits as follows:

$$\text{LCL} = \mu - 3\sigma_{\bar{x}} \tag{12.2}$$

$$\text{UCL} = \mu + 3\sigma_{\bar{x}} \tag{12.3}$$

[6] Note that this is another example of variability pooling. Choosing $n > 1$ tightens our estimate of $\bar{x}$ and therefore reduces our chances of reacting to random noise in the system.

If we observe a sample mean outside the range between LCL and UCL, then this observation is designated as assignable-cause variation. In the casting example charted in Figure 12.1, such a deviation occurred at sample 22. This might have been caused by defective inputs (e.g., steel or sand), machine problems (e.g., in the mold, the packing process, the pouring process), or operator error. SPC does not tell us why the deviation occurred—only that it is sufficiently unusual to warrant further investigation.

Other criteria besides points outside the control limits are sometimes used to signal out-of-control conditions. For instance, the occurrence of several points in a row above (or below) the target mean is frequently used to spot a potential shift in the process mean. In Figure 12.1, sample 37 is out of control. But unlike the out-of-control point at sample 22, this point is accompanied by an unusual run of above-average observations in samples 35 to 40. This is strong evidence that the cause of the problem is not unique to sample 37, but instead is due to something in the casting process itself that has caused the mean diameter to increase. Other criteria based on multiple samples, such as rules that look for trends (e.g., high followed by low followed by high again), are also used with control charts to spot assignable-cause variation.

It is important to note that because a process is in statistical control does *not* necessarily mean that it is **capable** (i.e., able to meet process specifications with regularity). For instance, suppose in our casting example that for reasons of functionality we require the hole diameter to be between a **lower specification level (LSL)** and an **upper specification level (USL).** Whether or not the process is capable of achieving these levels depends on how they compare with the **lower and upper natural tolerance limits,** which are defined as

$$\text{LNTL} = \mu - 3\sigma \tag{12.4}$$

$$\text{UNTL} = \mu + 3\sigma \tag{12.5}$$

Note that LNTL and UNTL are limits on the diameter of individual holes, while the LCL and UCL are limits on the average diameter of samples. Moreover, note that LNTL and UNTL are internally determined by the process itself, while LSL and USL are externally determined by performance requirements.

Let us consider some illustrative cases. The natural tolerance limits are given by $\text{LNTL} = \mu - 3\sigma = 10 - 3(0.025) = 9.925$ and $\text{LNTL} = \mu + 3\sigma = 10 + 3(0.025) = 10.075$. Suppose that the specification levels are given by LSL = LSL1 = 9.975 and USL = USL1 = 10.025. It is apparent from Figure 12.2 that the casting process will produce a large fraction of nonconforming parts. To be precise, if hole diameters are normally distributed, then

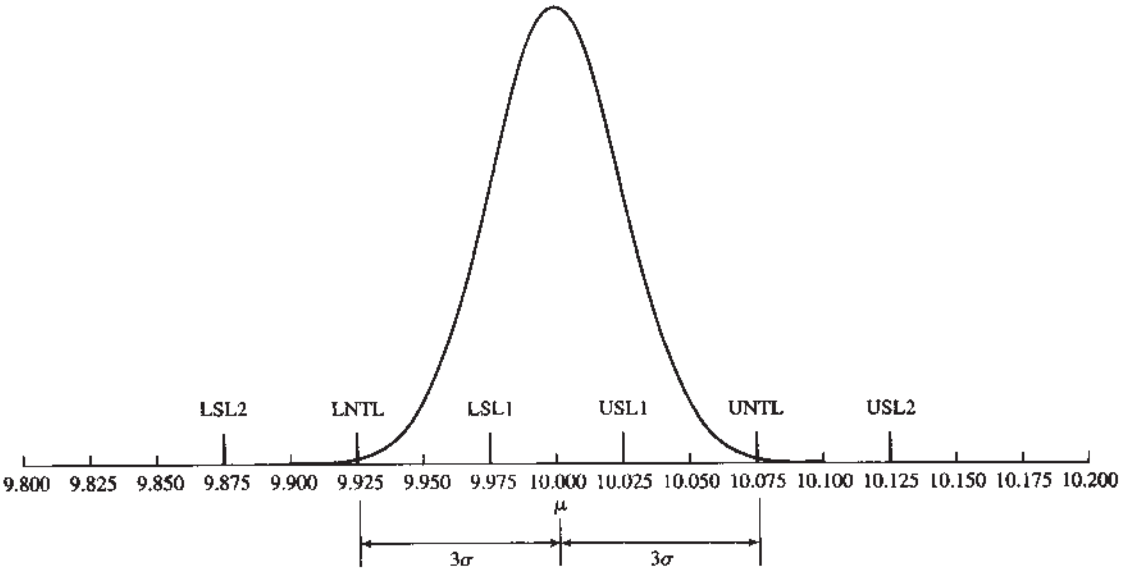
$$\begin{aligned} P(9.975 \le X \le 10.025) &= P\left(\frac{9.975 - 10}{0.025} \le Z \le \frac{10.025 - 10}{0.025}\right) \\ &= P(-1 \le Z \le 1) = \Phi(-1) + 1 - \Phi(1) \\ &= 0.1587 + 1 - 0.8413 \\ &= 0.3174 \end{aligned}$$

This means that almost 32 percent will fail to meet specification levels.

Suppose instead that the specification levels are given by LSL = LSL2 = 9.875 and USL = USL2 = 10.125. Since the natural tolerance limits lie well within this range, we would expect very few nonconforming castings. Indeed, repeating the calculation above for these limits shows that the fraction of nonconforming parts will be 0.00000057.

FIGURE 12.2

Process capability comparing specification limits to natural tolerance limits



A measure of capability is the **process capability index,** which is defined as

$$C_{pk} = \frac{Z_{\min}}{3} \tag{12.6}$$

where

$$Z_{\min} = \min\{-Z_{\mathrm{LSL}}, Z_{\mathrm{USL}}\} \tag{12.7}$$

and

$$Z_{\mathrm{LSL}} = \frac{\mathrm{LSL} - \mu}{\sigma} \tag{12.8}$$

$$Z_{\mathrm{USL}} = \frac{\mathrm{USL} - \mu}{\sigma} \tag{12.9}$$

The minimum acceptable value of C_{pk} is generally considered to be one. Note that in the above examples, $C_{pk} = 1/3$ for (LSL1, USL1), but $C_{pk} = 5/3$ for (LSL2, USL2). Note that C_{pk} is sensitive to both variability (σ) and asymmetry (i.e., a process mean that is not centered between USL and LSL). Hence, it gives us a simple quantitative measure of how capable a process is of meeting its performance specifications.

Of course, a host of details needs to be addressed to implement an effective SPC chart. We have glossed over the original estimates of μ and σ; in practice, there are a variety of ways to collect these from observable data. We also need to select the sample size n to be large enough to prevent reacting to random fluctuations but not so large that it masks assignable-cause variation. The frequency with which we sample must be chosen to balance the cost of sampling with the sensitivity of the monitoring.

12.3.3 SPC Extensions

The $\bar{x}$ chart discussed is only one type of SPC chart. Many variations have been proposed to meet the needs of a wide variety of quality assurance situations. A few that are particularly useful in manufacturing management include these:

1. *Range (R charts).* An $\bar{x}$ chart requires process variability (that is, σ) to be in control in order for the control limits to be valid. Therefore, it is common to monitor this variability by charting the range of the samples. If $x_1, x_2, \ldots, x_n$ are the measurements (e.g., hole diameters) in a sample of size n, then the range is the difference between the largest and smallest observations

$$R = x_{\max} - x_{\min} \tag{12.10}$$

Each sample yields a range, which can be plotted on a chart. Using past data to estimate the mean and standard deviation of R, denoted by $\bar{R}$ and σ_R, we can set the control limits for the R chart as

$$\text{LCL} = \bar{R} - 3\sigma_R \tag{12.11}$$

$$\text{UCL} = \bar{R} + 3\sigma_R \tag{12.12}$$

If the R chart does not indicate out-of-control situations, then this is a sign that the variability in the process is sufficiently stable to apply an $\bar{x}$ chart. Often, $\bar{x}$ and R charts are tracked simultaneously to watch for changes in either the mean or the variance of the underlying process.

2. *Fraction nonconforming (p charts).* An alternative to charting a physical measure, as we do in an $\bar{x}$ chart, is to track the fraction of items in periodic samples that fail to meet quality standards. Note that these standards could be quantitative (e.g., a hole diameter is within specified bounds) or qualitative (e.g., a wine is approved by a taster). If each item independently has probability p of being defective, then the variance of the fraction of nonconforming items in a sample of size n is given by $p(1 - p)/n$. Therefore, if we estimate the fraction of nonconforming items from past data, we can express the control limits for the p chart as

$$\text{LCL} = p - 3\sqrt{\frac{p(1-p)}{n}} \tag{12.13}$$

$$\text{UCL} = p + 3\sqrt{\frac{p(1-p)}{n}} \tag{12.14}$$

3. *Nonquality applications.* The basic control chart procedure can be used to track almost any process subject to variability. For example, we describe a procedure for statistical throughput control in Chapter 14, which monitors the output from a process in order to determine whether it is on track to attain a specified production quota. Another nonquality application of control charts is in due date quoting, which we discuss in Chapter 15. The basic idea is to attach a safety lead time to the estimated cycle time and then track customer service (e.g., as percentage delivered on time). If the system goes out of control, then this is a signal to adjust the safety lead time.

The power and flexibility of control charts make them extremely useful in monitoring all sorts of processes where variability is present. Since, as we have stressed repeatedly in this book, virtually all manufacturing processes involve variability, SPC techniques are a fundamental part of the tool kit of the modern manufacturing manager.

12.4 Quality and Operations

Closely related to variability as a link between quality and operations is cost. However, there is some disagreement about just how this link works. Here are two distinct views:

1. *Cost increases with quality.* This is the traditional industrial engineering view, which holds that achieving higher external quality requires more intense inspection, more rejects, and more expensive materials and processes. Since customers' willingness to pay for additional quality diminishes with the level of quality, this view leads to the "optimal defect level" arguments common to industrial engineering textbooks in the past.
2. *Cost decreases with quality.* This is the more recent TQM view, espoused using phrases such as *quality is free* (Crosby 1979) or *the hidden factory;* it holds that the material and labor savings from doing things right the first time more than offset the cost of the quality improvements. This view supports the zero-defects and continual-improvement goals of JIT.

Neither view is universally correct. If improving quality of a particular product means replacing a copper component with a gold one, then cost does increase with quality. Where this is the case, it makes sense to ask whether the market is willing to pay for, or will even notice, the improvement. On the other hand, if quality improvement is a matter of shifting some responsibility for inspection from end-of-line testing to individual machine operators, it is entirely possible that the reduction in rework, scrap, and inspection costs will more than offset the implementation cost. Ultimately, what matters is which view is appropriate for assessing the costs and consequences of a specific quality improvement. This is crucial for deciding which policies should be pursued while making continual improvements, and which should be tempered by the market.

In the next discussion and examples, we rely on the factory physics framework to evaluate the impacts of quality on operations and the impacts of operations on quality. Our intent is not so much to provide specific numerical estimates of the cost of quality—the range of situations that arise in industry is too varied to permit comprehensive treatment of this nature—but rather to broaden and extend the intuition we developed for the behavior of manufacturing systéms in Part II to incorporate quality considerations.

12.4.1 Quality Supports Operations

In Chapter 9 we presented two manufacturing laws that are central to understanding the impact of quality on plant operations, the variability law and the utilization law. These can be paraphrased as follows:

1. Variability causes congestion.
2. Congestion increases nonlinearly with utilization.

In practice, quality problems are one of the largest and most common causes of variability. Additionally, by causing work to be done over (either as rework or as replacements for scrapped parts), quality problems often end up increasing the utilization of workstations. By affecting both variability and capacity, quality problems can have extreme operational consequences.

The Effect of Rework on a Single Machine. To get a feel for how quality affects utilization and variability, let us consider a simple single-machine example. The machine receives parts at a rate of one every three minutes. Processing times have a mean and standard deviation of t_0 and σ_0 minute, respectively, so that the CV of the natural process time is $c_0 = \sigma_0/t_0$. However, with probability p, a given part is defective. We assume that the quality check is integral to the processing, and therefore whether the part is defective is immediately known upon its completion. If it is defective, it must be reworked, which requires another processing time with mean t_0 and standard deviation σ_0 and again has probability p of failing to produce a good part. The machine continues reworking the part until a good one is produced. We define the total time it takes to produce a good part to be the **effective processing time.**

Letting T_e represent the (random) effective processing time of a part, we can compute the mean t_e, variance σ_e^2, and squared coefficient of variation (SCV) c_e^2 of this time, as well as the utilization of the machine u, as follows:

$$t_e = E[T_e] = \frac{t_0}{1-p} \tag{12.15}$$

$$\sigma_e^2 = \text{Var}(T_e) = \frac{\sigma_0^2}{1-p} + \frac{pt_0^2}{(1-p)^2} \tag{12.16}$$

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = \frac{(1-p)\sigma_0^2 + pt_0^2}{t_0^2} = c_0^2 + p(1-c_0^2) \tag{12.17}$$

$$u = \frac{1}{3}t_e = \frac{t_0}{3(1-p)} \tag{12.18}$$

We can draw the following conclusions from this example:

1. *Utilization increases nonlinearly with rework rate.* This occurs because the mean time to process a job increases with the expected number of passes, while the arrival rate of new jobs remains constant. At some point, the added workload due to rework will overwhelm the station. In this example, Equation (12.18) shows that for $p > 1 - t_0/3$, utilization exceeds one, indicating that the system does not have enough capacity to keep up with both new arrivals and rework jobs over the long run.

2. *Variance of process time, given by σ_e^2, increases with rework rate.* The reason, of course, is that the more likely a job is to make multiple passes through the machine, the more unpredictable its completion time becomes.

3. *Variability of process time, as measured by the SCV, may increase or decrease with rework rate, depending on the natural variability of the process.* Although both the variance and the mean of the effective process time always increase with the rework rate, the variance does not always increase faster than the mean. Hence the SCV, which is the ratio of variance to mean, can increase or decrease. We can see from Equation (12.17) that c_e^2 increases in p if $c_0^2 < 1$, decreases in p if $c_0^2 > 1$, and is constant in p if $c_0^2 = 1$. The intuition behind this is that the effects of variability pooling (which happens when we sum the process times of repeated passes) become large enough when $c_0^2 > 1$ to cause the SCV of effective process times to decrease in p.

We can use these specific results for a single machine with rework to motivate some general observations about the effect of rework on the cycle time and lead time of a process. Since both the mean and the variance of effective process time increase with rework rate, we can invoke the lead time law of Chapter 9 to conclude that the lead time required to achieve a given service level also increases the rework rate.

The effect of rework on cycle time is not so obvious, however. The fact that the SCV of effective process time can go down when rework increases, may give the impression that rework might actually reduce cycle time. But this is not the case. The reason is that increasing rework increases utilization, which is a first-order effect on cycle time that outweighs the second-order effect from a possible reduction in variability. Hence, even in processes with high natural variability, increasing rework will inflate the mean cycle time. Moreover, because it also increases the variance of total processing time per job and the variance of the time to wait in queue, increasing rework also inflates the standard deviation of cycle time. These cycle time effects represent general observations about the impact of rework, as we summarize in the following manufacturing law.

Law (Rework): *For a given throughput level, rework increases both the mean and standard deviation of the cycle time of a process.*

To give an illustration of this law, suppose the previously mentioned station is fed by a moderately variable arrival process (that is, $c_a = 1$) but has deterministic processing times such that $t_0 = 1$ and $c_0 = 0$. Then, using Kingman's model of a workstation introduced in Chapter 8, the cycle time at the station can be expressed as a function of p as

$$\begin{aligned} \text{CT} &= \frac{c_a^2 + c_e^2}{2} \frac{u}{1-u} t_e + t_e \\ &= \frac{1+p}{2} \frac{1/(3(1-p))}{1-1/(3(1-p))} \frac{1}{1-p} + \frac{1}{1-p} \end{aligned}$$

Figure 12.3 plots cycle time versus rework rate. This plot shows that cycle time grows nonlinearly toward infinity as p approaches 2/3, the point at which rework reduces the effective capacity of the system below the arrival rate.

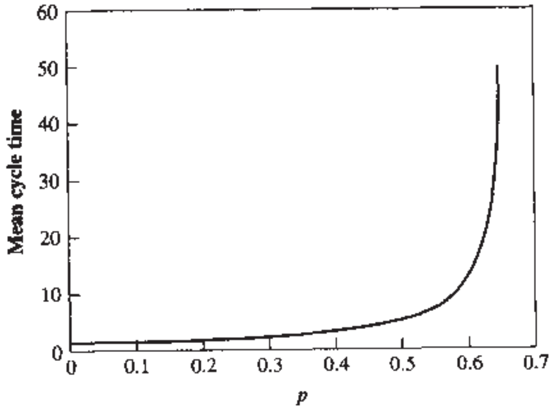
Effect of Rework on a CONWIP Line. Of course, station level measures such as utilization, variability, and cycle time are only indirect measures; what we really care about is the throughput, WIP, and cycle time of a line. To illustrate the rework law in a line, consider the CONWIP line depicted in Figure 12.4. Processing times are two-thirds hour for machines 1, 2, and 4 and one hour for machine 3 (the bottleneck). All processing times are deterministic (that is, $c_e^2 = 0$). However, machine 2 is subject to rework. As in the previous example, we assume that each job that is processed must be reprocessed with probability p. Hence, as in the previous example, the mean effective processing time on machine 2 is given by

$$t_e(2) = \frac{2/3}{1-p}$$

We assume that the line has unlimited raw materials, so the only source of variability is rework.

FIGURE 12.3

Cycle time as a function of rework rate



Because even this simple line is too complex to permit convenient analysis (the single-machine example was messy enough!), we turn to computer simulation to estimate the performance measures for various values of p and different WIP levels. Figures 12.5 and 12.6 summarize our simulation results.

When $p = 0$ (no rework), the system behaves as the best case we studied in Chapter 7. Thus, we can apply the formulas derived there to characterize the throughput-versus-WIP and cycle-time-versus-WIP curves. Note that without rework, the bottleneck rate r_b is one job per hour, and the raw process time T_0 is $r_b T_0 = 3$ hours. Hence, the critical WIP level is 3 jobs. At this WIP level, maximum throughput (1 job per hour) and minimum cycle time (three hours) are attained.

When $p = 1/3$, the mean effective process time on machine 2 is $t_e(2) = 1$, the bottleneck rate. Thus, r_b is not changed, but T_0 increases to 3.33 hours. This means that as WIP approaches infinity, full throughput of one job per hour will be attained. Our simulation indicates that virtually full throughput is attained at a WIP level of about 10 jobs—more than three times the WIP level required in the no-rework case. At a WIP level of 10 jobs, the average cycle time is roughly 10 hours—also three times the ideal level of the

Figure 12.4

A CONWIP line with rework

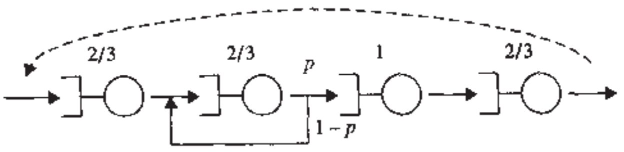


Figure 12.5

Throughput versus WIP for different rework rates

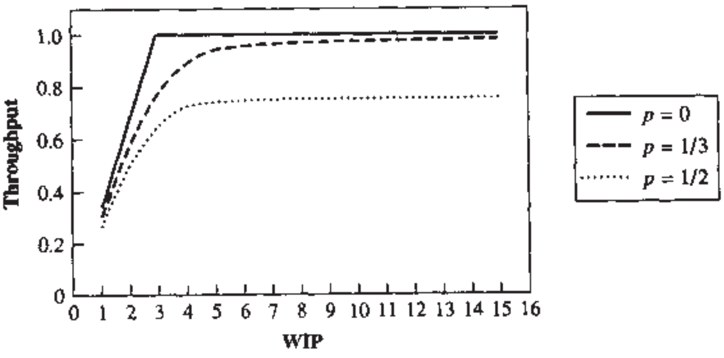
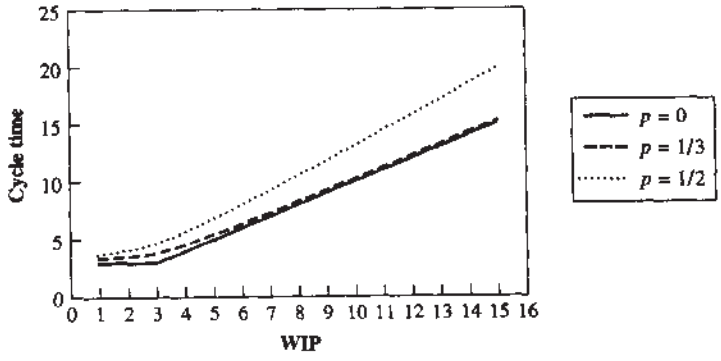


Figure 12.6

Cycle time versus WIP for different rework rates



no-rework case. The implication here is that the primary effect of rework when $p = 1/3$ is to transform a line that behaved as the best case to one approaching the practical worst case. This illustrates the rework law in action with regard to the mean cycle time.

When $p = 1/2$, the mean effective process time on machine 2 is $t_e(2) = 4/3$, which makes it the bottleneck. Thus, even with infinite WIP, we cannot achieve throughput above $r_b = 3/4$ job per hour. As expected, Figure 12.5 shows substantially reduced throughput at all WIP levels. Figure 12.6 shows that cycle times are longer, as a consequence of the reduced capacity at machine 2, at all WIP levels. Moreover, because the bottleneck rate has been decreased, the cycle time curve increases with WIP at a faster rate than in the previous two cases.

The simulation model enables us to keep track of other line statistics. Of particular interest is the standard deviation of cycle time. Recall that the lead time law implies that if we quote customer lead times to achieve a specified service level (probability of on-time delivery), then lead times are an increasing function of both average cycle time and the standard deviation of cycle time. Larger standard deviation of cycle time means we will have to quote longer lead times, and consequently must hold items in finished goods inventory longer, to compensate for the variable production rate. As Figure 12.7 shows, the standard deviation of cycle time increases in the rework rate. Moreover, it increases in the WIP level (as there is more WIP in the line to cause random queueing delays at the stations). Since, as we noted, rework requires additional WIP in the line to achieve a given throughput level, this effect tends to aggravate further the cycle time variability problem. This is an illustration of the rework law with regard to variance of cycle time.

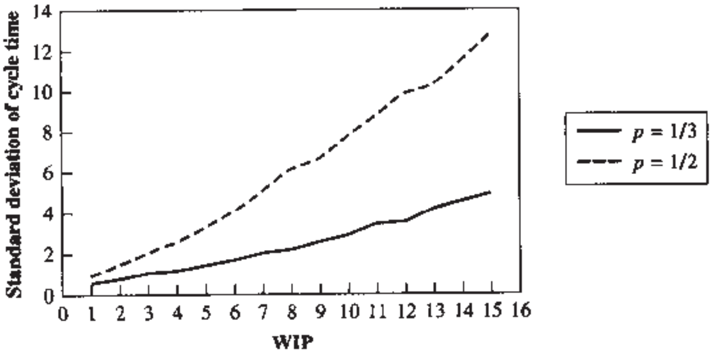
The results of Figures 12.5, 12.6, and 12.7 imply the following about the operations and cost impacts of quality problems.

1. *Throughput effects.* If the rework is high enough to cause a resource to become a bottleneck (or, even worse, the rework problem is *on* the bottleneck resource), it can substantially alter the capacity of the line. Where this is the case, a quality improvement can facilitate an increase in throughput. The increased revenue from such an improvement can *vastly* exceed the cost of improving quality in the line.

2. *WIP effects.* Rework on a nonbottleneck resource, even one that has plenty of spare capacity, increases variability in the line, thereby requiring higher WIP (and cycle time) to attain a given level of throughput. Thus, reductions in rework can facilitate reductions in WIP. Although the cost savings from such a change are not likely to be as large as the revenue enhancement from a capacity increase, they can be significant relative to the cost of achieving the improvement.

FIGURE 12.7

Standard deviation of cycle time versus WIP for different rework rates



3. *Lead time effects.* By decreasing capacity and increasing variability, rework problems necessitate additional WIP in the line and hence lead to longer average cycle times. These problems also increase the variability of cycle times and hence lead to either longer quoted lead times or poorer service to the customer. The competitive advantage of shorter lead times and more reliable delivery, achieved via a reduction in rework, is difficult to quantify precisely but can be of substantial strategic importance.

Further Observations. We conclude our discussion on the operations impacts of quality problems with some observations that go beyond the preceding examples.

To begin, we note that *the longer the rework loop, the more pronounced the consequences.* In the two examples above, we represented rework as a second pass through a single machine. In practice, rework is frequently much more involved than this. A defective part may have to loop back through several stations in the line in order to be corrected. When this is the case, rework affects the capacity and variability of effective processing time on several stations. Additionally, because each pass through the rework loop adds even more time than in the single-machine rework loop case, the effect on the standard deviation of cycle time tends to be larger. As a result, the consequences of the rework law become even more pronounced as the length of the rework loop grows.

Because rework has such a disruptive effect on a production line, manufacturing managers are frequently tempted to set up separate rework lines. Such an approach does prevent defective parts from sapping capacity and inflating variability in the main line. However, it does this by installing extra capacity somewhere else, which costs money, takes up space, and does little to eliminate the inflation of the mean and standard deviation of cycle time caused by rework. Even worse, such an approach can serve to sweep quality problems under the rug. Shunting defective parts to a separate line makes them someone else's responsibility. Making a line responsible for correcting its own problems fosters greater awareness of the causes and effects of quality problems. If such awareness can lead to quicker detection of problems, it can shorten the rework loop and mitigate the consequences. If it can lead to ways to avoid the defects in the first place, then truly major improvements can be achieved. Consequently, despite the short-term appeal of separate rework lines, it is probably better in the long run to avoid them and strive for more fundamental quality improvements.

In many manufacturing environments, internal quality problems lead to **scrap**—that is, yield loss—rather than rework, either because the defect cannot be corrected or because it is not economical to do so. Thus, it is important to point out that *scrap has similar effects to rework.* From an operations standpoint, scrapped parts are essentially identical to reworked parts that must be processed again from the beginning of the line. In this sense, scrap is the most extreme form of rework and therefore has the same effects we observed for rework, only more so.

A difference between scrap and rework, however, lies in the method used to compensate. While separate lines can be used for rework, they make no sense as a remedy for scrap. Instead, most manufacturing systems perform some form of job size inflation as protection against yield loss. (We first discussed this approach in Chapter 3 in the context of MRP but will review it again here in the context of quality and operations.) The most obvious approach is to divide the desired quantity by the expected yield rate. For example, if we have an order for 90 parts and the yield rate is 90 percent (i.e., a 10 percent scrap rate), then we could release

$$\frac{90}{0.9} = 100$$

units. Then if 10 percent are lost to scrap, we will have 90 good parts to ship to the customer.

This approach would be fine if the scrap rate were truly a deterministic constant (i.e., we *always* lose 10 percent). But in virtually all real situations, the scrap rate for a given job is a random quantity; it might range from 0 to 100 percent. When this is the case, it is not at all clear that inflating by the expected yield rate is the best approach. For instance, in the previous example, suppose the *expected* scrap rate is 90 percent, but what really happens is that 90 percent of the time the yield for a given job is 100 percent (no yield loss) and the other 10 percent of the time it is 0 percent (catastrophic yield loss). If we inflate by dividing the amount demanded by the customer by 0.9, then 90 percent of the time we will wind up with excess and the other 10 percent of the time we will be short. In this extreme case, job inflation does not improve customer service at all!

When too little good product finishes to fill an order, we must start additional parts and wait for them to finish before we can ship the entire amount to the customer. That is, it is similar to a rework loop that encompasses the entire line. Unless we have built in substantial lead time to the customer, this is likely to result in a late delivery. The costs to the firm are the (hard to quantify) cost of lost customer goodwill and the cost of disrupting the line to rush the makeup order through the line.

On the other hand, when low yield loss results in more good product finishing than required to fill an order, the excess will go into finished goods inventory (FGI) and be used to fill future orders. The cost to the firm is that incurred to hold the extra inventory in FGI. Of course, if all products are customized and cannot be used against future demand, the extra inventory will amount to scrap.

At any rate, there is no reason to expect the cost of being short on an order by n units to be equal to that of being over it by n units. In most cases, the cost of being short exceeds that of being over. Hence, from a cost minimization standpoint, it might make sense to inflate by *more* than the expected yield loss. For instance, in a situation where yield varies between 80 and 100 percent, we might divide the amount demanded by 0.85 instead of 0.9, so that we release 106 parts instead of 100 to cover an order of 90. This would allow us to ship on time as long as the yield loss was not greater than 15 percent.

But in cases where yield loss is frequently all or nothing (e.g., we get either 100 good parts or none from a release quantity of 100), inflating job size is generally futile. (We would have to start an entire second job of 100 parts to make up for the catastrophic failure of the first batch.) A more practical alternative is to carry safety stock in finished goods inventory; for example, we try to carry n jobs' worth of FGI, where n is the number of scrapped jobs we want to be able to cover. In a system with many products, this can require considerable (expensive) inventory.

The unavoidable conclusion is that scrap loss caused by variable yields is costly and disruptive. The more variable the yields, the more difficult it is to mitigate the effect with inflated job sizes or safety stocks. Thus, in the long term, the best option is to strive to minimize or eliminate scrap and rework.

12.4.2 Operations Supports Quality

The previous subsection stressed that better quality promotes better operations. Happily, the reverse is also frequently true. As pointed out frequently in the JIT literature, to the extent that tighter operations management leads to less WIP (i.e., shorter queues), it aids in the detection of quality problems and facilitates tracing them to their source.

Specifically, suppose that there tends to be a great deal of WIP between a point in a production line that causes defects and the point where these defects are detected. The defects might be caused by a machine early in the line because it has imperceptibly gone "out of control" but not be detected until an end-of-line (EOL) test. By the time a defect is detected at the EOL test, it is likely that all the parts that have been produced by the upstream machine are similarly defective. If the line has a high WIP level in it, scrap loss could be large. If the line has little WIP, scrap loss is likely to be much less.

Of course, in the real world, causes and detection of defects are considerably more complex and varied than this. There are likely to be many sources of potential defects, some of which have never been encountered before—or at least, for which there is no institutional memory. Detection of defects can occur at many places in the line, both at formal inspection points and as a result of informal observations elsewhere. While these realities serve to make understanding and managing quality a challenge, they do not alter the main point: High WIP levels tend to aggravate scrap loss by increasing the time, and hence number of items produced, between the cause and the detection of a defect.

Example: A Defect Detection

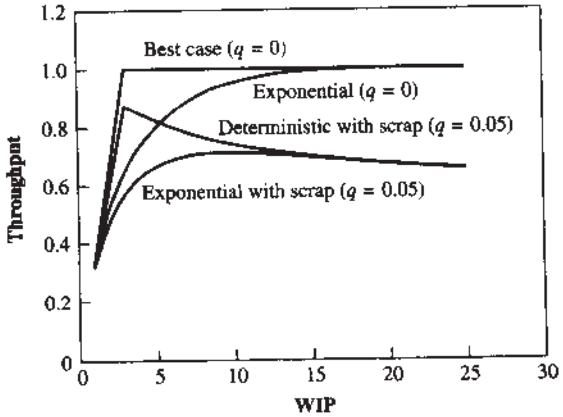
Consider again the CONWIP line depicted in Figure 12.4, only this time suppose that the rework rate at machine 2 is zero. Instead, suppose that each time a job is processed on machine 1, there is a probability q that this machine goes out of control and produces bad parts until it is fixed. However, the out-of-control status of machine 1 can only be inferred by detecting the bad parts, which does not occur until after the parts have been processed at machine 4. Each time a defective part is detected, we assume that machine 1 is corrected instantly. But all the parts that have been produced on machine 1 between the time it went out of control and the time the defect was detected at machine 4 will be defective and must be scrapped at the end of the line.

Figure 12.8 illustrates the curve of throughput (of good parts only) versus WIP for four cases of this example. First, when $q = 0$ (no quality problems) and all processing times are deterministic, we get the familiar best-case curve. Second, for comparison, we plot throughput versus WIP when $q = 0$ but processing times are exponential (i.e., they have CVs of 1). Here, throughput increases with WIP, reaching nearly maximum output at around 15 jobs. Note that this curve is somewhat better than (i.e., lies above) the practical worst case due to the imbalance in the line.

However, when $q = 0.05$ and processing times are deterministic, throughput increases and then declines with WIP. The reason, of course, is that for high WIP levels, the increased scrap loss outweighs the higher production rate it promotes. The maximum throughput occurs at a WIP level of three jobs, the critical WIP level. When $q = 0.05$

FIGURE 12.8

Throughput versus WIP in a system with scrap loss



and processing times are exponential, throughput again increases and then decreases, with maximum throughput being achieved at a WIP level of nine jobs. Notice that while we can make up for the variability induced by random processing times by maintaining a high WIP level (for example, 15 jobs), the variability due to scrap loss is only aggravated by more WIP. So instead of putting more WIP in the system to compensate, we must *reduce* the WIP level to mitigate this second form of variability and thereby maximize throughput. Metaphorically speaking, this is like lowering the water to cover the rocks. Obviously, metaphors have their limits.

It is our guess that in real life, throughput-versus-WIP curves frequently do exhibit this increasing-then-decreasing type of behavior, not only because of poor quality detection but also because high WIP levels make it harder to keep track of jobs, so that more time is wasted locating jobs and finding places to put them between processes. Moreover, more WIP leads to more chances for damage. In general, we can conclude that better operations (i.e., tighter WIP control) leads to better quality (less scrap loss) and hence higher throughput (better operations again). This is a simple illustration of the fact that quality and operations are mutually supportive and therefore can be jointly exploited to promote a cycle of continual improvement.

12.5 Quality and the Supply Chain

Total quality management refers to quality outside, as well as inside, the walls of the plant. Under the topic of vendor certification (e.g., ISO 9000), the TQM literature frequently mentions the **supply chain:** the network of plants and vendors that supply raw material, components, and services to one another. Almost all plants today rely on outside suppliers for at least some of the inputs to their manufacturing process. Indeed, the tendency in recent years has been toward *vertical deintegration* through outsourcing of an increasing percentage of manufactured components.

When significant portions of a finished product come from outside sources, it is clear that internal, and perhaps external, quality at the plant can depend critically on these inputs. As computer programmers say, "garbage in, garbage out." (Or as farmers say, "you can't make a silk purse out of a sow's ear.") Whatever the metaphor, the point is that a TQM program must address the issue of purchased parts if it is to be effective. Vendor certification, working with fewer vendors, using more than price to choose between vendors, and establishing quality assurance procedures as close to the front of the line as possible—all are options for improving purchased part quality. The choice and character of these policies obviously depend on the setting. We refer the reader to the previously cited TQM references for more in-depth discussion.

Just as internal scrap and rework problems can have significant operations consequences, quality problems from outside suppliers can have strong impacts on plant performance. First, any defects in purchased parts that find their way into the production process to cause scrap or rework problems will affect operations in the fashion we have discussed. However, even if defective purchased parts are screened out before they reach the line, either at the supplier plant or at the receiving dock, these quality problems can still have negative operational effects. The reason is that they serve to inflate the *variability of delivery time.* If scrap or rework problems at the supplier plant cause some orders to be delivered late, or if some orders must be sent back because quality problems were detected upon receipt, the effective delivery time (i.e., the time between submission of a purchase order and receipt of acceptable parts) will not be regular and predictable.

12.5.1 A Safety Lead Time Example

To appreciate the effects of variable delivery times for purchased parts, consider the following example. A plant has decided to purchase a particular part from one of two suppliers on a lot-for-lot basis. That is, the company will not buy the part in bulk and stock it at the plant, but instead will bring in just the quantities needed to satisfy the production schedule. If the part is late, the schedule will be disrupted and customer deliveries may be delayed. Therefore, management chooses to build a certain amount of **safety lead time** into the purchasing lead time. The result is that, on average, parts will arrive somewhat early and wait in raw materials inventory until they are needed at the line. The key question is, How much safety lead time is required?

Figure 12.9 depicts the probability density functions (pdf's) for the delivery time from the two candidate suppliers. Both suppliers have mean delivery times of 10 days. However, deliveries from supplier 2 are much more variable than those from supplier 1 (perhaps because supplier 2 does not have sound OM and TQM systems in place). As a result, to be 95 percent certain that an order will arrive on time (i.e., when required by the production schedule), parts must be ordered with a lead time of 14 days from supplier 1 or a lead time of 23 days from supplier 2 (see Figure 12.10). The additional lead time is required for supplier 2 to make up for the variability in delivery time. Notice that this implies that an average order from supplier 1 will wait in raw materials inventory for $14 - 10 = 4$ days, while an average order from supplier 2 will wait in raw materials inventory for $23 - 10 = 13$ days—an increase of 225 percent. From Little's law, we know that raw materials inventory will also be 225 percent larger if we purchase from supplier 2 rather than from supplier 1.

12.5.2 Purchased Parts in an Assembly System

The effects of delivery time variability become even more pronounced when assemblies are considered. In many manufacturing environments, a number of components are purchased from different suppliers for assembly into a final product. To avoid a schedule disruption, *all* the components must be available on time. Because of this, the amount of safety lead time needed to achieve the same probability of being able to start on time is larger than it would be if there were only a single purchased component.

To see how this works, consider an example in which a product is assembled from 10 components, all of which are purchased from separate vendors and have the same

Figure 12.9

Effect of delivery time variability on purchasing lead times

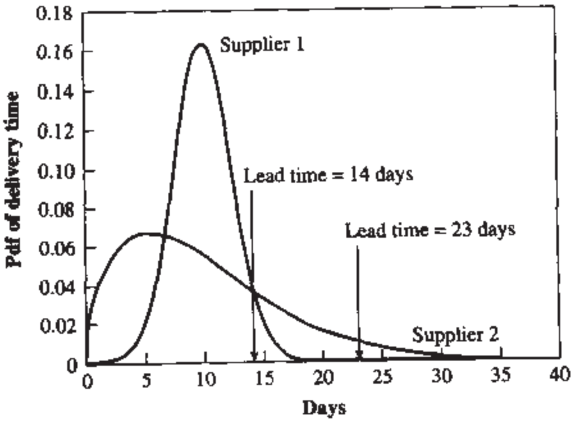
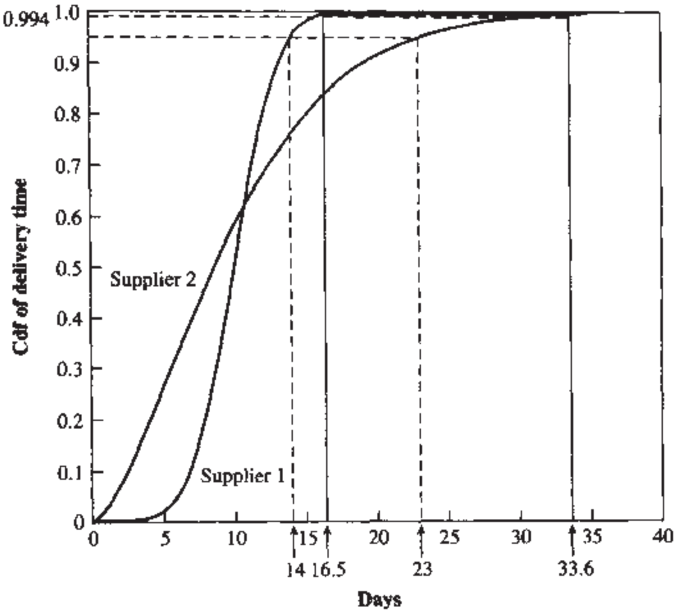


FIGURE 12.10

Setting safety lead times in multiple-component systems



distribution (i.e., mean and variance) of delivery time. Since the parts are identical with regard to their delivery characteristics, it is sensible to choose the same purchasing lead time for all. Suppose this is done as in the previous single-component example so that each component has a 95 percent chance of being received on time. Assuming delivery times of the different components to be independent, the probability that all are on time is given by the *product* of the individual on-time probabilities

$$\text{Prob}\{\text{all 10 components arrive on time}\} = (0.95)^{10} = 0.5987$$

Assembly will be able to start on time less than 60 percent of the time!

Obviously, the plant needs longer lead times and higher individual on-time probabilities to achieve the desired 95 percent likelihood of having all components in when required by the schedule. Specifically, if we let p represent the on-time percentage for a single part, we want

$$p^{10} = 0.95$$

or

$$p = 0.95^{1/10} = 0.9949$$

To ensure that the entire set of parts is available 95 percent of the time, each individual part must be available 99.49 percent of the time.

To see the operations effects of this, consider Figure 12.10, which shows the cumulative distribution function (cdf) of the delivery times from supplier 1.[7] This curve gives the probability that the delivery time is less than or equal to t for all values of t. For a single component to be available 95 percent of the time, a purchasing lead time of 14 days (i.e., a safety lead time of four days) is sufficient. However, for a single component to be available 99.49 percent of the time, in order to support the 10-component assembly system, a purchasing lead time of 16.3 days (i.e., a safety lead time of 6.3 days) is needed. Thus, purchased parts will reside in raw materials inventory for an additional 2.3 days

[7] The cdf is simply the area under the pdf shown in Figure 12.9 from 0 to t.

on average in the multicomponent assembly system, and therefore the raw materials inventories will be increased by a corresponding amount.

Since multiple-component systems require high individual on-time probabilities, the tails of the delivery time distributions are critical. For instance, the purchasing lead time required for supplier 2 in Figure 12.10 to achieve a 99.49 percent probability of on-time delivery is 33.6 days. Recall that in the single-component case, there was a difference of nine days between the required lead times for suppliers 1 and 2 (that is, 14 days for supplier 1 and 23 days for supplier 2). In the 10-component case, there is a difference of $33.6 - 16.3 = 17.3$ days. The conclusion is that reliable suppliers are *extremely* important to efficient operation of an assembly system that involves multiple purchased parts.

12.5.3 Vendor Selection and Management

The preceding discussion has something (though far from everything) to say about the problem of supplier selection. To see what, suppose components are purchased from two separate suppliers. Each has a probability p of delivering on time, so that the probability of receiving both parts on time is p^2. Now, further suppose that both parts could be purchased from a single vendor. If that vendor could provide better on-time performance than p^2 for the combined shipments, then, all other things being equal, it would be better to switch to the single vendor. Even if the purchasing cost is higher when using the single vendor, the savings in inventory and schedule disruption costs may justify the switch. Having fewer vendors providing multiple parts might produce better on-time performance than having many vendors providing single parts, for these reasons:

1. Purchases become a larger percentage, and therefore a higher-priority piece, of the supplier's business.
2. The purchasing department can keep better track of suppliers (by knowing about special circumstances that would alter the usual purchasing lead times, by being able to place "reminder" phone calls, etc.) if there are fewer of them.

The insights from these simplified examples extend to more realistic systems. Obviously, in the real world, suppliers do not have identical delivery time distributions, nor are the costs of the different components necessarily similar. For these reasons, it may make sense to set the on-time delivery probabilities differently for different components. An inexpensive component (e.g., a resistor) should probably have a very high on-time probability because the inventory cost of achieving it is low.[8] An expensive component (e.g., a cathode-ray tube display) should have a relatively lower on-time probability, in order to reduce its safety lead time and hence average inventory level. The general idea is that if a schedule disruption is going to occur, it ought to be due to a $500 cathode-ray tube, not a 2-cent resistor.

Formal algorithms exist for computing appropriate safety lead times in assembly systems with multiple nonidentical purchased components (see Hopp and Spearman 1993). But whether we use algorithms or less rigorous methods to establish safety lead times for the individual components, the result will be to set an on-time probability for each component. As our previous discussion of Figure 12.9 illustrated, for a fixed

[8] Actually, for really inexpensive items that are used with some regularity, it makes sense to simply order them in bulk and stock them on site to ensure that they are virtually never out of stock. However, this advice does not apply to bulky materials (e.g., packaging) for which the cost of storage space and handling makes large on-site stocks uneconomical.

on-time probability, safety lead time and raw materials inventory are both increasing in the variance of supplier delivery time. Moreover, as we observed in Figure 12.10, the more independent suppliers we order from, the higher the individual on-time probabilities required to support a given probability of maintaining schedule.

This discussion can be thought of as a quick factory physics interpretation of the JIT view on vendoring. The JIT literature routinely suggests certifying a smaller number of vendors, precisely because low delivery time variance is needed to support just-in-time deliveries. Indeed, Toyota has evolved a very extensive system of working with its suppliers that goes well beyond simple certification—to the point of sending in advisers to set up the "Toyota system," which addresses both quality and operations, in the supplier's plant. The goal is to nurture suppliers that effectively support Toyota's operation *and* are efficient enough to remain economically viable partners over the long term.

12.6 Conclusions

Quality is a broad and varied subject, which ranges from definitions of customer needs to analytical measurement and maintenance tools. In this chapter, we have tried to give a sense of this range and have suggested references for the interested reader to consult for additional depth. In keeping with the factory physics framework of this book, we have concentrated primarily on the relationship between quality and operations and have shown that the two are intimately related in a variety of ways. Specifically, we have argued the following:

1. *Good quality supports good operations.* Reducing recycle and/or scrap serves to increase capacity and decrease congestion. Thus, better quality control—through tighter control of inputs, mistake prevention, and earlier detection—facilitates increased throughput and reduced WIP, cycle time, and customer lead time.
2. *Good operations supports quality improvement.* Reducing WIP—via better scheduling, pull mechanisms for shop floor control, or (although it is hardly an imaginative option) capacity increases—serves to reduce the amount of product generated between the cause of a defect and its detection. This has the potential to reduce the scrap and rework rate and to help identify the root causes of quality problems.
3. *Good quality at the supplier level promotes good operations and quality at the plant level.* A supplier plant with fewer scrap, rework, and external quality problems will make more reliable deliveries. This enables a customer plant to use shorter purchasing lead times for these parts (e.g., just-in-time becomes a possibility), to carry smaller raw materials inventories, and to avoid frequent schedule disruption.

Based on these discussions, we conclude that both quality and operations are integral parts of a sound manufacturing management strategy. One cannot reasonably consider one without the other. Hence, perhaps we should really view total quality management more in terms of *quality of management* than *management of quality*.

Study Questions

1. Why is quality so difficult to define? Provide your own definition for a specific operation of your choosing.
2. Give three major ways that good internal quality can promote good external quality.

3. Using the following definition of the cost of quality

> *Quality costs are defined as any expenditures on manufacturing or service in excess of those that would have been incurred if the product had been built or the service had been performed exactly right the first time.*
>
> Garvin (1988, 78)

identify the costs associated with each of the following types of quality problems:

a. A flow line with a single-product family where defects detected at any station are scrapped.

b. A flow line with a single-product family where defects detected at any station are reworked through a portion of the line.

c. A cutting machine where bit breakage destroys the part in production and brings the machine down for repair.

d. Steel burners for a kitchen range that are coated with a porcelain that cracks off after a small amount of use in the field.

e. A minivan whose springs for holding open the hatchback are prone to failure.

f. A cheap battery in new cars and light trucks that fails after about 18 months when the warranty period is 12 months.

4. For each of the following examples, would you expect cost to increase or decrease with quality? Explain your reasoning.

a. An automobile manufacturer increases expected battery life by installing more expensive batteries in new cars.

b. A publisher reduces the number of errors in newly published books by assigning extra proofreaders.

c. A steel rolling mill improves the consistency of its galvanizing process through installation of a more sophisticated monitoring system (i.e., that measures temperature, pH, etc., at various points in the chemical bath).

d. A manufacturer of high-voltage switches eliminates quality inspection of metal castings after certifying the supplier from which they are purchased.

e. An automobile manufacturer repairs an obvious defect (e.g., a defective paint job) after the warranty period has expired.

5. What quality implications could setup time reduction have in a manufacturing line?

6. How might improved internal quality make scheduling a production system easier?

7. Why do the operational consequences of rework become more severe as the length of the rework loop increases?

8. How are the operational consequences of rework similar to those of scrap? How are they different?

9. Why is it important to detect quality problems as early in the line as possible?

Problems

1. Manov Steel, Inc., has a rolling mill that produces sheet steel with a nominal thickness of 0.125 inch. Suppose that the specification limits are given by LSL $= 0.120$ and USL $= 0.130$ inch. Based on historical data, the actual thickness of a random sheet produced by the mill is normally distributed with mean and standard deviation of $\mu = 0.125$ and $\sigma = 0.0025$.

a. What are the lower and upper natural tolerance limits (LNTL and UNTL) for individual sheets of steel?

b. What are the lower and upper control limits (LSL and USL) if we use a control chart that plots the average thickness of samples of size $n = 4$?

c. What will be the percentage nonconforming, given the above values for (LNTL, UNTL) and (LSL, USL)? What is the process capability index C_{pk}? Do you consider this process capable of meeting its performance specifications?

d. Suppose that the process mean suddenly shifts from 0.125 to 0.1275. What happens to the process capability index C_{pk} and the percentage nonconforming?

e. Under the conditions of d, what is the probability that the $\bar{x}$ chart specified in b will detect an out-of-control signal on the first sample after the change in process mean?

2. A purchasing agent has requested quotes for valve gaskets with diameters of 3.0 ± 0.018 in. SPC studies of three suppliers have indicated that their processes are in statistical control and produce measurements that are normally distributed with the following statistics:

Supplier 1: $\mu = 3$ inches	$\sigma = 0.009$ inch
Supplier 2: $\mu = 3$ inches	$\sigma = 0.0044$ inch
Supplier 3: $\mu = 2.99$ inches	$\sigma = 0.003$ inch

Assuming that all suppliers offer the same price and delivery reliability/flexibility, which supplier should the agent purchase from? Explain your reasoning.

3. Consider a single machine that requires one hour to process parts. With probability p, a given part must be reworked, which requires a second one-hour pass through the machine. However, all parts are guaranteed to be good after a second pass, so none go through more than twice.

   a. Compute the mean and variance of the effective processing time on this machine as a function of p.

   b. Use your answer from a to compute the squared coefficient of variation (SCV) of the effective processing times. Is it an increasing function of p? Explain.

4. Suppose the machine in Problem 1 is part of a two-station line, in which it feeds a second machine that has processing times with a mean of 1.2 hours and SCV of 1. Jobs arrive to the line at a rate of 0.8 job per hour with an arrival SCV of 1.

   a. Compute the expected cycle time in the line when $p = 0.1$.

   b. Compute the expected cycle time in the line when $p = 0.2$.

   c. What effects does rework have on cycle time, and how do these differ in a and b?

5. Suppose a cellular telephone plant purchases electronic components from various suppliers. For one particular component, the plant has a choice between two suppliers: Supplier 1 has delivery lead times with a mean of 15 days and a standard deviation of 1 day, while supplier 2 has delivery lead times with a mean of 15 days and a standard deviation of 5 days. Both suppliers can be assumed to have normally distributed lead times.

   a. Assuming that the cellular plant purchases the component on a lot-for-lot basis and wants to be 99 percent certain that the component is in stock when needed by the production schedule, how many days of lead time are needed if supplier 1 is used? Supplier 2?

   b. How many days will a typical component purchased from supplier 1 wait in inventory before being used? From supplier 2? How might this information be used to justify using supplier 1 even if it charges a higher price?

   c. Suppose that the cellular plant purchases (on a lot-for-lot basis) 100 parts from different suppliers, all of which have delivery times like those of supplier 1. Assuming all components are assigned the same lead time, what lead times are required to ensure that *all* components are in stock when required by the schedule? How does your answer change if all suppliers have lead times like those of supplier 2?

   d. How would your answer to a be affected if, instead of ordering lot for lot, the cellular plant ordered the particular component in batches corresponding to five days' worth of production?

6. Consider a workstation that machines castings into switch housings. The castings are purchased from a vendor and are prone to material defects. If all goes well, machining (including load and unload time) requires 15 minutes, and the SCV of natural processing time (due to variability in the time it takes the operator to load and start the machine) is 0.1. However, two types of defect in the castings can disrupt the process.

One type of defect (a flaw) causes the casting to crack during machining. When this happens, the casting is scrapped at the end of the operation and another casting is machined. About 15 percent of castings have this first type of defect.

A second type of defect (a hard spot) causes the cutting bit to break. When this happens, the machine must be shut down, must wait for a repair technician to arrive, must be examined for damage, and must have its bit replaced. The whole process takes an average of two hours, but is quite variable (i.e., the standard deviation of the repair time is also two hours). Furthermore, since the casting must be scrapped, another one must be machined to replace it once the repair is complete. About five percent of castings have this second type of defect.

a. Compute the mean and SCV of effective process time (i.e., the time it takes to machine a good housing). (*Hint:* Use Equations (12.15) and (12.17) to consider the effects of the first type of defect, and consult Table 8.1 for formulas to address the second type of defect. *Question:* Should stoppages due to the second type of defect be modeled as preempt or nonpreempt outages?)

b. How does your answer to *a* change if the defect percentages are reversed (that is, five percent of castings have the first type of defect, while 15 percent have the second type)? What does this say about the relative disruptiveness of the two types of defects?

c. Suppose that by feeding the castings through the cutting tool more slowly, we could ensure that the second type of defect does not cause bit breakage. Under this policy, castings with the second type of defect will be scrapped, but will not cause any machine downtime (i.e., they become identical to the first type of defect). However, this increases the average time to machine a casting without defects from 15 minutes to t minutes. What is the maximum value of t for which the slower feed speed achieves at least as much capacity as the original situation in *a*?

d. Which workstation would you rather manage, that in *a* (i.e., fast feeds and bit breakages) or that in *c* (i.e., slow speeds, resulting in machining times equal to your answer to *c*, and no bit breakages)? (*Hint:* How do the effective SCVs of the two cases compare?)

PART

III Principles in Practice

In matters of style, swim with the current;
In matters of principle, stand like a rock.
Thomas Jefferson

CHAPTER

13 A Pull Planning Framework

We think in generalities, we live in detail.
Alfred North Whitehead

13.1 Introduction

Recall that we began this book by stating that the three critical elements of an operations management education are

1. Basics
2. Intuition
3. Synthesis

We spent almost all Parts I and II on the first two items. For instance, the tools and terminology introduced in Part I (for example, EOQ, (Q, r), BOM, MPS) and the measures of variability (e.g., coefficient of variation) and elementary queueing concepts presented in Part II are *basics* of fundamental importance to the manufacturing manager. The insights from traditional inventory models, MRP, and JIT we observed in Part I and the factory physics relationships among throughput, WIP, cycle time, and variability we developed in Part II are key components of sound *intuition* for making good operating decisions.

But, with the exception of a bit of integration of the contrasting perspectives of operations and behavioral science in Chapter 11 and the pervasive aspects of quality presented in Chapter 12, we have devoted almost no time to the third item, **synthesis.** We are now ready to fill in this important gap by establishing a framework for applying the principles from Parts I and II to real manufacturing problems.

Our approach is based on two premises:

1. Problems at different levels of the organization require different levels of detail, modeling assumptions, and planning frequency.
2. Planning and analysis tools must be consistent across levels.

The first premise motivates us to use separate tools for separate problems. Unfortunately, using different tools and procedures throughout the system can easily bring us into conflict with the second premise. Because of the potential for inconsistency, it is not uncommon to find planning tools in industry that have been extended across applications for which they are ill suited. For instance, we once worked in a plant that used a scheduling tool that calculated detailed, *minute-by-minute* production on each machine in the plant to generate *two-year* aggregate production plans. Although this tool may have been reasonable for short-term planning (e.g., a day or a week), it was far too cumbersome to run for long-term purposes (the data input and debugging alone took an entire week!). Moreover, it was so inaccurate beyond a few weeks into the future that the schedule, so painfully obtained, was virtually ignored on the plant floor.

To develop methods that are both well suited to their specific application *and* mutually consistent across applications, we recommend the following steps in developing a planning framework:

1. **Divide the overall system appropriately.** Different planning methods for different portions of the process, different product categories, different planning horizons, different shifts, etc., can be used. The key is to find a set of divisions that make each piece manageable, but still allow integration.
2. **Identify links between the divisions.** For instance, if production plans for two products with a shared process center are made separately, they should be linked via the capacity of the shared process. If we use different tools to plan production requirements over different time horizons, we should make sure that the plans are consistent with regard to their assumptions about capacity, product mix, staffing, etc.
3. **Use feedback to enforce consistency.** All analysis, planning, and control tools make use of estimated parameters (e.g., capacity, machine speeds, yields, failure and repair rates, demand rates, and many others). As the system runs, we should continually update our knowledge of these values. Rather than allow the inputs to the various tools to be estimated in an ad hoc, uncoordinated fashion, we should explicitly make use of our updated knowledge to force tools to make use of timely, consistent information.

In the remainder of this chapter, we preview a planning framework that is consistent with these principles, as well as the factory physics principles presented earlier. We do not pretend that this framework is the only one that is consistent with these principles. Rather, we offer it as one approach and try to present the issues involved at the various levels from a sufficiently broad perspective as to allow room for customization to specific manufacturing environments. Subsequent chapters in Part III will flesh out the major components of this framework in greater detail.

13.2 Disaggregation

The first step in developing a planning structure is to break down the various decision problems into manageable subproblems. This can be done explicitly, through the development of a formal planning hierarchy, as we will discuss. Or it can be done implicitly by addressing the various decisions piecemeal with different models and assumptions. Regardless of the level of foresight, some form of disaggregation *will* be done, since all real-world production systems are too complex to address with a single model.

13.2.1 Time Scales in Production Planning

One of the most important dimensions along which manufacturing systems are typically broken down is that of *time*. The primary reason for this is that manufacturing decisions differ greatly with regard to the length of time over which their consequences persist.

For example, the construction of a new plant will affect a firm's position for years or even decades, while the effects of selecting a particular part to work on at a particular workstation may evaporate within hours or even minutes. This makes it essential to use different **planning horizons** in the decision-making process. Since the decision to construct a new plant will influence operations for years, we must forecast these effects years into the future in order to make a reasonable decision. Hence, the planning horizon should be long for this problem. Clearly, we do not need to look nearly so far into the future to evaluate the decision of what to work on at a workstation, so this problem will have a short planning horizon.

The appropriate length of the planning horizon also varies across industries and levels of the organization. Some industries, oil and long-distance telephone, for example, routinely make use of horizons as long as several decades because the consequences of their business decisions persist this long. Within a given company, longer time horizons are generally used at the corporate office, which is responsible for long-range business planning, than at the plant where day-to-day execution decisions are made.

In this book we focus primarily on decisions relevant to running a plant, and we divide planning horizons in this context into **long, intermediate, and short**. At the plant level, a long planning horizon can range from one to five years with two years being typical. An intermediate planning horizon can range from a week to a year, with a month being typical. A short time horizon can range from an hour to a week, with a day being typical.

Table 13.1 lists various manufacturing decisions that are made over long, intermediate, and short planning horizons. Notice that in general, long-range decisions address **strategy,** by considering such questions as what to make, how to make it, how to finance

TABLE 13.1 Strategy, Tactics, and Control Decisions

Time Horizon	Length	Representative Decisions
Long term (strategy)	Year to decades	Financial decisions Marketing strategies Product designs Process technology decisions Capacity decisions Facility locations Supplier contracts Personnel development programs Plant control policies Quality assurance policies
Intermediate term (tactics)	Week to year	Work scheduling Staffing assignments Preventive maintenance Sales promotions Purchasing decisions
Short term (control)	Hour to week	Material flow control Worker assignments Machine setup decisions Process control Quality compliance decisions Emergency equipment repairs

it, how to sell it, where to make it, and where to get materials and general principles for operating the system. Intermediate-range decisions address **tactics,** by determining what to work on, who will work on it, what actions will be taken to maintain the equipment, what products will be pushed by sales, and so on. These tactical decisions must be made within the physical and logical constraints established by the strategic long-range decisions. Finally, short-range decisions address **control,** by moving material and workers, adjusting processes and equipment, and taking whatever actions are required to ensure that the system continues to function toward its goal. Both the long-term strategic and intermediate-range tactical decisions establish the constraints within which these control decisions must be made.

Different planning horizons imply different **regeneration frequencies.** A long-range decision that is based on information extending years into the future does not need to be reconsidered very often, because the estimates about what will happen this far into the future do not change very fast. For instance, while it is a good thing for a plant to reevaluate what products it should be making, this is not a decision that should be reconsidered every week. Typically, long-range problems are considered on a quarterly to annual basis, with very long-range issues (e.g., what business should we be in?) being considered even less frequently. Intermediate-range problems are reconsidered on roughly a weekly to monthly basis. Short-range problems are reconsidered on a real-time to daily basis. Of course, these are merely typical values, and considerable variation occurs across firms and decision problems.

In addition to differing with respect to regeneration frequency, problems with different planning horizons differ with respect to the required *level of detail.* In general, the shorter the planning horizon, the greater the amount of detail required in modeling and data collection. For instance, if we are making a long-term strategic capacity decision about what size plant to build, we do not need to know very much about the routings that parts will take. It may be enough to have a rough estimate of how much time each part will require of each process, in order to estimate capacity requirements. However, at the intermediate tactical level, we need more information about these routings, for instance, which specific machines will be visited, in order to determine whether a given schedule is actually feasible with respect to customer requirements. Finally, at the short-term control level, we may need to know a great deal about part routings, including whether or not a given part requires rework or other special attention, in order to guide parts through the system.

A good analogy for this strategy/tactics/control distinction is mapmaking. Long-term problems are like long-distance travel. We require a map that covers a large amount of distance, but not in great detail. A map that shows only major highways may be adequate for our needs. Likewise, a long-term decision problem requires a tool that covers a large amount of time (i.e., long planning horizon), but not in great detail. In contrast, short-term problems are like short-distance travel. We require a map that does not cover much distance, but gives lots of details about what it does cover. A map showing city streets, or even individual buildings, may be appropriate. Analogously, for a short-term decision problem, we require a tool that does not cover much time (i.e., short planning horizon), but gives considerable detail about what it does cover.

13.2.2 Other Dimensions of Disaggregation

In addition to time, there are several other dimensions along which the production planning and control problem is typically broken down. Because modern factories are large and complex, it is frequently impossible to consider the plant as a whole when one is making specific decisions. The following are three dimensions that can be used to break the plant into more manageable pieces for analysis and management:

1. **Processes.** Traditionally, many plants were organized according to physical manufacturing processes. Operations such as casting, milling, grinding, drilling, and heat treat were performed in separate departments in distinct locations and under different management. While such process organization has become less popular in the wake of the JIT revolution, with its flow-oriented cellular layouts, process divisions still exist. For instance, casting is operationally very different, and sometimes physically distant, from rolling in a steel mill. Likewise, mass lamination of copper and fiberglass cores in large presses is distinct—physically, operationally, and logistically—from the circuitizing process in which circuitry is etched into the copper in a photo-optical/chemical flow line process. In such situations, it frequently makes sense to assign separate managers to the different processes. It may also be reasonable to use different planning, scheduling, and control procedures.

2. **Products.** Although plants dedicated to a single product exist (e.g., a polystyrene plant), most plants today make multiple products. Indeed, the pressure to compete via variety and customization has probably served to increase the average number of different products produced by an average plant. For instance, it is not uncommon to find a plant with 20,000 distinct part numbers (i.e., counting finished products and subcomponents). Because it is difficult, under these conditions, to consider part numbers individually, many manufacturing plants aggregate part numbers into coarser categories for planning and management purposes.

One form of aggregation is to lump parts with identical routings together. Typically, there are many fewer routings through the plant than there are part numbers. For instance, a printed-circuit board plant, which produces several thousand different circuit boards, may have only two **basic routings** (e.g., for small and large boards). Frequently, however, the actual number of routings can be substantially larger than the number of basic routings if one counts minor variations (e.g., extra test steps, vendoring of individual operations, and gold plating of contact surfaces) in the basic routing. For planning, it is generally desirable to keep the number of "official" routings to a minimum by ignoring minor variations.

In systems with significant setup times, aggregation by routing may be going too far. For instance, a particular routing in a circuit board line may produce 1,000 different circuit boards. However, there may be only four different thicknesses of copper. Since the speed of the conveyor must be changed with thickness (to ensure proper etching), a setup involving lost capacity must be made whenever the line switches thicknesses. In addition, the 1,000 boards may require three different dies for punching rectangular holes in the boards. Whenever the line switches between boards requiring different dies, a setup is incurred. If all possible combinations of copper thickness and die requirement are represented in the 1,000 boards, then there are $4 \times 3 = 12$ distinct **product families** within the routing. This definition of *family* ensures that there are no significant setups *within* families but there may be setups *between* families. As we will discuss in Chapter 15, setups have important ramifications for scheduling. For this reason, aggregation of products by family can often simplify the planning process without oversimplifying it.

3. **People.** There are a host of ways that a factory's workforce can be broken down: labor versus management, union versus nonunion, factory floor versus staff support, permanent versus temporary, departments (e.g., manufacturing, production control, engineering, personnel), shifts, and so on. In a large plant, the personnel organization scheme can be almost as complex as the machinery. While a detailed discussion of workforce organization is largely beyond the scope of this book—we touched on some

of the issues involved in Chapter 11—we feel it is important to point out the logistical implications of such organizations. For instance, having separate managers for different processes or shifts can lead to a lack of coordination. Relying on temporary workers to facilitate a varying workforce can decrease the institutional memory, and possibly the skill level, of the organization. Rigidly adhering to job descriptions can preclude opportunities for cross-training and flexibility within the system. As we stressed in Chapter 11, the effectiveness of a manufacturing system is very much a function of its workforce. While it will always be necessary to classify workers into different categories for purposes of training, compensation, and communication, it is important to remember that we are not necessarily constrained to follow the procedures of the past. By taking a perspective that is sensitive to logistics and people, a good manager will seek effective personnel policies that support both.

13.2.3 Coordination

There is nothing revolutionary about the previous discussion about separating decision problems along the dimensions of time, process, product, or people. For instance, virtually every manufacturing operation in the world does some sort of long-, intermediate-, and short-range decision making. What distinguishes a good system from a bad one is not whether it makes such a breakdown, but how well the resulting subproblems are solved and, especially, how well they are coordinated with one another. We will examine the subproblems in some detail in the remaining chapters of Part III. For now, we begin addressing the issue of coordination by means of an illustration.

The problem of what parts to make at what times is addressed at the long-, intermediate-, and short-term levels. Over the long term, we must worry about rough volumes and product mix in order to be able to plan for capacity and staffing. Over the intermediate term, we must develop a somewhat more detailed production plan, in order to procure materials, line up vendors, and rationally negotiate customer contracts. Over the short term, we must establish and execute a detailed work schedule that controls what happens at each process center. The basic essence of all three problems is the same; only the time frame is different. Hence, it seems obvious that the decisions made at the three different levels should be consistent, at least in expectation, with one another. As one might expect, this is easier to say than to do.

When we generate a long-range production plan, giving the quantity of each part to produce in various time buckets (typically months or quarters), we cannot possibly consider the production process in enough detail to determine the exact number of machine setups that will be required. However, when we develop an intermediate-range production schedule, we must compute the required number of setups, because otherwise we cannot determine whether the schedule is feasible with respect to capacity. Therefore, for the long-range plan to be consistent with the intermediate-range plan, we should make sure that the long-range planning tool subtracts an amount from the capacity of each process center that corresponds to an anticipated average number of setups. To ensure this over time, we should track the actual number of setups and adjust the long-range planning accordingly.

A similar link is needed between the intermediate- and short-term plans. When we generate an intermediate-range production schedule, we cannot anticipate all the variations in material flow that will occur in the actual production process. Machines may fail, operators may call in sick, process or quality problems may arise—none of which can be foreseen. However, at the short-range level, when we are planning minute by minute what to work on, we must consider what machines are down, what workers are absent, and many other factors affecting the current status of the plant. The result will be that actual production activities will never match planned ones exactly. Therefore, for the short-range activities to be able to generate outputs that are consistent, at least on average, with planned requirements, the intermediate-range planning tool must contain some form of buffer capacity or buffer lead time to accommodate randomness. Buffer capacity might be provided in the form of the "two-shifting" we discussed in Chapter 4 on JIT. Buffer lead times are simply additions to the times we quote to customers to allow for unanticipated delays in the factory.

Next we will discuss other links between planning levels in the context of specific problems. However, since the reader is certain to encounter planning tools and procedures other than those discussed in this book, we have raised the issue of establishing links as a general principle. The main point is that the various levels can and should be addressed with different tools and assumptions, but linked via simple mechanisms such as those discussed previously.

13.3 Forecasting

The starting point of virtually all production planning systems is forecasting. This is because the consequences of manufacturing planning decisions almost always depend on the future. A decision that looks good now may turn out later to be terrible. But since no one has a crystal ball with which to predict the future, the best we can do is to make use of whatever information is available in the present to choose the policies that we predict will be successful in the future.

Obviously, dependence on the future is not unique to manufacturing. The success or failure of government policies is heavily influenced by future parameters, such as interest rates, economic growth, inflation, and unemployment. Profitability of insurance companies depends on future liabilities, which are in turn a function of such unpredictable things as natural disasters. Cash flow in oil companies is governed by future success in drilling ventures. In cases like these, where the effectiveness of current decisions depends on uncertain outcomes in the future, decision makers generally rely on some type of **forecasting** to generate expectations of the future in order to evaluate alternate policies.

Because there are many approaches one can use to predict the future, forecasting is a large and varied field. One basic distinction is between methods of

1. Qualitative forecasting
2. Quantitative forecasting

Qualitative forecasting methods attempt to develop likely future scenarios by using the expertise of people, rather than precise mathematical models. One structured method for eliciting forecasts from experts is **Delphi.** In Delphi, experts are queried about some future subject, for instance, the likely introduction date of a new technology. This is usually done in written form, but can be done orally. The responses are tabulated and returned to the panel of experts, who reconsider and respond again, to the original and possibly some new questions as well. The process can be repeated several times, until consensus is reached or the respondents have stabilized in their answers. Delphi and techniques like it are useful for long-term forecasting where the future depends on the past in very complex ways. Technological forecasts, where predicting highly uncertain breakthroughs is at the core of the exercise, frequently use this type of approach. Martino (1983) summarizes a variety of qualitative forecasting methods in this context.

Quantitative forecasting methods are based on the assumption that the future can be predicted by using numerical measures of the past in some kind of mathematical model. There are two basic classes of quantitative forecasting models:

1. **Causal models** predict a future parameter (e.g., demand for a product) as a function of other parameters (e.g., interest rates, growth in GNP, housing starts).
2. **Time series models** predict a future parameter (e.g. demand for a product) as a function of past values of that parameter (e.g., historical demand).

Because we cannot hope to provide a comprehensive overview of forecasting, we will restrict our attention to those techniques that have the greatest relevance to operations management (OM). Specifically, because operational decisions are primarily concerned with problems having planning horizons of less than two years, the long-term techniques of qualitative forecasting are not widely used in OM situations. Therefore, we will focus on quantitative methods. Furthermore, because time series models are simple to use and have direct applicability (in a nonforecasting context) to the production tracking module, we will devote most of our attention to these.

Before we cover specific techniques, we note the following well-known laws of forecasting:

First law of forecasting: *Forecasts are always wrong!*

Second law of forecasting: *Detailed forecasts are worse than aggregate forecasts!*

Third law of forecasting: *The further into the future, the less reliable the forecast will be!*

No matter how qualified the expert or how sophisticated the model, perfect prediction of the future is simply not possible; hence the first law. Furthermore, by the concept of variability pooling, an aggregate forecast (e.g., of a product family) will exhibit less variability than a detailed forecast (e.g., of an individual product); hence the second law. Finally, the further out one goes, the greater the potential for qualitative changes (e.g., the competition introduces an important new product) that completely invalidate whatever forecasting approach we use; hence the third law.

We do not mean by these laws to disparage the idea of forecasting altogether. On the contrary, the whole notion of a planning hierarchy is premised on forecasting. There is simply no way to sensibly make decisions of how much capacity to install, how large a workforce to maintain, or how much inventory to stock without some estimate of future demand. *But* since our estimate is likely to be approximate at best, we should strive to make these decisions as robust as possible with respect to errors in the forecast. For instance, using equipment and plant layouts that enable accommodation of new products, changes in volume, and shifts in product mix, sometimes referred to as **agile manufacturing,** can greatly reduce the consequences of forecasting errors. Similarly, cross-training of workers and adaptable workforce scheduling policies can substantially increase flexibility. Finally, as we noted in Part II, shortening manufacturing cycle times can reduce dependence on forecasts.

13.3.1 Causal Forecasting

In a causal forecast, we attempt to explain the behavior of an uncertain future parameter in terms of other, observable or at least more predictable, parameters. For instance, if we are trying to evaluate the economics of opening a new fast-food outlet at a given location, we need a forecast of demand. Possible predictors of demand include population and number of competitor fast-food restaurants within some distance of the location. By collecting data on demand, population, and competition for existing comparable restaurants, we can use statistics to estimate constants in a model.

The most commonly used model is the simple linear model, of the form

$$Y = b_0 + b_1X_1 + b_2X_2 + \cdots + b_mX_m \tag{13.1}$$

where Y represents the parameter to be predicted (demand) and the X_i variables are the predictive parameters (population and competition). The b_i values are constants that must be statistically estimated from data.

This technique for fitting a function to data is called **regression analysis;** many computer packages, including all major spreadsheet programs, are available for performing the necessary computations. The following example briefly illustrates how regression analysis can be used as a tool for causal forecasting.

Example: Mr. Forest's Cookies

An emerging cookie store franchise was in the process of evaluating sites for future outlets. Top management conjectured that the success of a store is strongly influenced by the number of people who live within five miles of it. Analysts collected this population data and annual sales data for 12 existing franchises, as summarized in Table 13.2.

To develop a model for predicting the sales of a new franchise from its five-mile-radius population, the analysts made use of regression analysis, which is a tool for finding the "best-fit" straight line through the data. They did this by choosing the **Regression** function in Excel, which produced the output shown in Figure 13.1. The three key numbers, marked in boldface, are as follows:

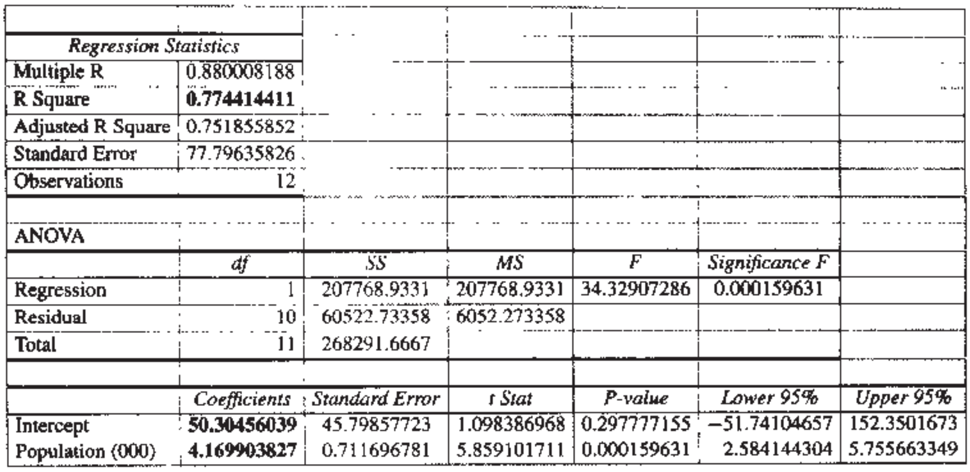
1. **Intercept coefficient,** which is the estimate of b_0 in Equation (13.1), or 50.30 (rounded to two decimals) for this problem. This coefficient represents the Y intercept of the straight line being fit through the data.
2. **X_1 coefficient,** or the estimate of b_1 in Equation (13.1), which is 4.17 for this problem. This coefficient represents the slope of the straight line being fit through the data. It is indicated as "Population (000)" in Figure 13.1.
3. **R square,** which represents the fraction of variation in the data that is explained by the regression line. If the data fit the regression line perfectly, R square will

TABLE 13.2 Mr. Forest's Cookies Franchise Data

Franchise	Population (000)	Sales ($000)
1	50	200
2	25	50
3	14	210
4	76	240
5	88	400
6	35	200
7	85	410
8	110	500
9	95	610
10	21	120
11	30	190
12	44	180

FIGURE 13.1

Excel regression analysis output

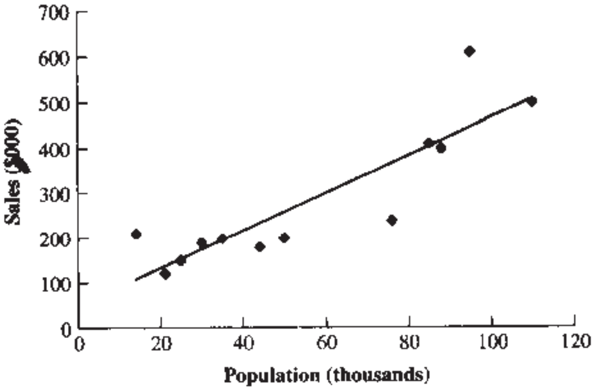


SUMMARY OUTPUT

Regression Statistics						
Multiple R	0.880008188					
R Square	**0.774414411**					
Adjusted R Square	0.751855852					
Standard Error	77.79635826					
Observations	12					
ANOVA						
	df	*SS*	*MS*	*F*	*Significance F*	
Regression	1	207768.9331	207768.9331	34.32907286	0.000159631	
Residual	10	60522.73358	6052.273358			
Total	11	268291.6667				
	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	**50.30456039**	45.79857723	1.098386968	0.297777155	−51.74104657	152.3501673
Population (000)	**4.169903827**	0.711696781	5.859101711	0.000159631	2.584144304	5.755663349

FIGURE 13.2

Fit of regression line to Mr. Forest's data



be one. The smaller R square is, the poorer the fit of the data to the regression line. In this case, R square is 0.77441441, which means that the fit is reasonably good, but hardly perfect. Excel also generates a plot of the data and the regression line, as shown in Figure 13.2, which allows us to visually examine how well the model fits the data.

Thus, the predictive model is given by

$$\text{Sales} = 50.30 + 4.17 \times \text{Population} \tag{13.2}$$

where sales are measured in thousands of dollars ($000) and population represents the five-mile-radius population in thousands. So a new franchise with a five-mile-radius population of 60 thousand would have predicted annual sales of

$$50.30 + 4.17(60) = \$300.5$$

The above equation is in thousands.

Judging from the results in Figures 13.1 and 13.2, the model appears reasonable for making rough predictions, provided that the population for the new franchise is between 15,000 and 110,000. Since the initial data set does not include populations outside this range, we have no basis for making predictions for populations smaller than 15,000 or larger than 110,000.

If the analysts for Mr. Forest want to develop a more refined model, they might consider adding other predictive variables, such as the average income of the five-mile-radius population, number of other cookie stores within a specified distance of the proposed location, and number of other retail establishments within walking distance of the proposed location. The general model of Equation (13.1), known as a **multiple regression model** (as opposed to a **simple regression model** that includes only a single predictive variable), allows such multiple predictors, as do the computer packages for performing the computations.

Packages such as Excel make the mechanics of regression simple. But full interpretation of the results requires knowledge of statistics. Given that statistics and regression are widely used throughout business—for marketing analysis, product design, personnel evaluation, forecasting, quality control, and process control—they are essential basics of a modern manager's skill set. Any good business statistics text can provide the necessary background in these important topics.

Although frequently useful, a causal model by itself cannot always enable us to make predictions about the future. For instance, if *next* month's demand for roofing materials, as seen by the manufacturer, depends on *last* month's housing starts (because of the time lag between the housing start and the replenishment purchase order placed on the manufacturer by the supplier), then the model requires only observable inputs and we can make a forecast directly. In contrast, if *next* month's demand for air conditioners depends on *next* month's average daily temperature, then we must forecast next month's temperature before we can predict demand. (Given the quality of long-term weather forecasts, it is not clear that such a causal model would be of much help, however.)

13.3.2 Time Series Forecasting

To predict a numerical parameter for which past results are a good indicator of future behavior, but where a strong cause-and-effect relationship is not available for constructing a causal model, a **time series model** is frequently used. Demand for a product often falls into this category, and therefore demand forecasting is one of the most common applications of this technique. The reason is that demand is a function of such factors as customer appeal, marketing effectiveness, and competition. Although these factors are difficult to model explicitly, they do tend to persist over time, so past demand is often a good predictor of future demand. What time series models do is to try to capture past trends and extrapolate them into the future.

Although there are many different time series models, the basic procedure is the same for all. We treat time in periods (e.g., months), labeled $i = 1, 2, \ldots, t$, where period t is the most recent data observation to be used in the forecast. We denote the actual observations by $A(i)$ and let the forecasts for periods $t + \tau$, $\tau = 1, 2, \ldots$, be represented by $f(t + \tau)$. As shown in Figure 13.3, a time series model takes as input the past observations $A(i)$, $i = 1, \ldots, t$ (for example, $A(i)$ could represent demand in month i, where t represents the most recent month for which data are available) and generates predictions for the future values $f(t+\tau)$, $\tau = 1, 2, \ldots$ (for example, $f(t+\tau)$ represents the forecasted demand for month $t + \tau$, which is τ months into the future).

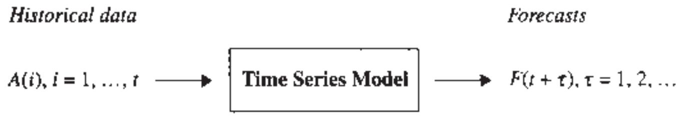


FIGURE 13.3

Basic structure of time series models

Toward this end, some models, including those discussed here, compute a **smoothed estimate** $F(t)$, which represents an estimate of the current position of the process under consideration, and a **smoothed trend** $T(t)$, which represents an estimate of the current trend of the process.

There are many different models that can perform this basic forecasting function; which is most appropriate depends on the specific application. Here we present four of the simplest and most common approaches. The **moving-average** model computes the forecast for the next period (and thereafter) as the average of the last m observations (where the user chooses the value of m). **Exponential smoothing** computes a smoothed estimate as a weighted average (where the user chooses the weights) of the most recent observation and the previous smoothed estimate. Like the moving-average model, simple exponential smoothing assumes no trend (i.e., upward or downward) in the data and therefore uses the smoothed estimate as the forecast for all future periods. **Exponential smoothing with a linear trend** estimates the smoothed estimate in a manner similar to exponential smoothing, but also computes a smoothed trend, or slope, in the data. Finally, **Winter's method** adds seasonal multipliers to the exponential smoothing with a linear trend model, in order to represent situations where demand exhibits seasonal behavior.

Moving Average. The simplest way to convert actual observations to forecasts is to simply average them. In doing this, we are implicitly assuming that there is no trend, so that $T(t) = 0$ for all t. We then compute the smoothed estimate as the simple average and use this average for all future forecasts, so that

$$F(t) = \frac{\sum_{i=1}^{t} A(i)}{t}$$

$$f(t+\tau) = F(t) \qquad \tau = 1, 2, \ldots$$

A potential problem with this approach is that it gives all past data equal weight regardless of their age. But demand data from three years ago may no longer be representative of future expectations. To capture the tendency for more recent data to be better correlated with future outcomes than old data are, virtually all time series models contain a mechanism for discounting old data. The simplest procedure for doing this is to throw data away beyond some point in the past. The time series model that does this is called the **moving-average** model, and it works in the same way as the simple average except that only the most recent m data points (where m is a parameter chosen by the user) are used in the average. Again, the trend is assumed to be zero, so $T(t) = 0$, and all future forecasts beyond the present are assumed to be equal to the current smoothed estimate:

$$F(t) = \frac{\sum_{i=t-m+1}^{t} A(i)}{m} \tag{13.3}$$

$$f(t+\tau) = F(t) \qquad \tau = 1, 2, \ldots \tag{13.4}$$

Notice that the choice of m will make a difference in how the moving-average method performs. A way to find an appropriate value for a particular situation is to try various values and see how well they predict already known data. For instance, suppose we have 20 months of past demand for a particular product, as shown in Table 13.3. At any time, we can pretend that we only have data up to that point and use our moving average to generate a forecast. If we set $m = 3$, then in period $t = 3$ we can compute the smoothed estimate as the average of the first three points, or

$$F(3) = \frac{10 + 12 + 12}{3} = 11.33$$

At time $t = 3$, our forecast for demand in period 4 (and beyond, since there is no trend) is $f(4) = F(3) = 11.33$. However, once we actually get to period 4 and make another observation of actual demand, our estimate becomes the average of the second, third, and fourth points, or

$$F(4) = \frac{12 + 12 + 11}{3} = 11.67$$

Now our forecast for period 5 (and beyond) is $f(5) = F(4) = 11.67$. Continuing in this manner, we can compute what our forecast would have been for $t = 4, \dots, 20$, as shown in Figure 13.3. We cannot make forecasts in periods 1, 2, and 3 because we need three data points before we can compute a three-period moving average.

If we change the number of periods in our moving average to $m = 5$, we can compute the smoothed estimate, and therefore the forecast, for periods $6, \dots, 20$, as shown in Table 13.3.

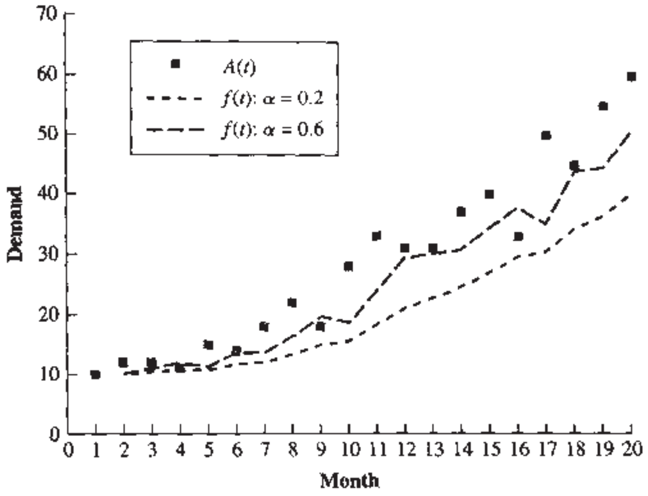
Which is better, $m = 3$ or $m = 5$? It is rather difficult to tell from Table 13.3. However, if we plot $A(t)$ and $f(t)$, we can see which model's forecast came closer to the

TABLE 13.3 Moving Averages with $m = 3$ and $m = 5$

		Forecast $f(t)$	
Month t	Demand $A(t)$	$m = 3$	$m = 5$
1	10	—	—
2	12	—	—
3	12	—	—
4	11	11.33	—
5	15	11.67	—
6	14	12.67	12.0
7	18	13.33	12.8
8	22	15.67	14.0
9	18	18.00	16.0
10	28	19.33	17.4
11	33	22.67	20.0
12	31	26.33	23.8
13	31	30.67	26.4
14	37	31.67	28.2
15	40	33.00	32.0
16	33	36.00	34.4
17	50	36.67	34.4
18	45	41.00	38.2
19	55	42.67	41.0
20	60	50.00	44.6

FIGURE 13.4

Moving average with $m = 3, 5$



actual observed values. As we see in Figure 13.4, both models tended to underestimate demand, with the $m = 5$ model performing worse. The reason for this underestimation is that the moving-average model assumes no upward or downward trend in the data. But we can see from the plots that these data clearly have an upward trend. Therefore, the moving average of past demand tends to be less than future demand. Since the model with $m = 5$ is even more heavily tied to past demand (because it includes more, and therefore older, points), it suffers from this tendency to a greater extent.

This example illustrates the following general conclusions about the moving-average model:

1. Higher values of m will make the model more stable, but less responsive, to changes in the process being forecast.
2. The model will tend to underestimate parameters with an increasing trend, and overestimate parameters with a decreasing trend.

We can address the problem of tracking a trend in the context of the moving-average model. For those familiar with regression analysis, the way this works is to estimate a slope for the last m data points via linear regression and then make the forecast equal to the smoothed estimate plus an extrapolation of this linear trend. However, there is another, easier way to introduce a linear trend into a different time series model. Next, we will pursue this approach after presenting another trendless model below.

Exponential Smoothing. Observe that the moving-average approach gives equal weight to each of the m most recent observations and no weight to observations older than these. Another way to discount old data points is to average the current smoothed estimate with the most recent data point. The result will be that the older the data point, the smaller the weight it receives in determining the forecast. We call this method **exponential smoothing,** and it works as follows. First, we assume, for now, that the trend is always zero, so $T(t) = 0$. Then we compute the smoothed estimate and forecast at time t as

$$F(t) = \alpha A(t) + (1 - \alpha)F(t - 1) \tag{13.5}$$

$$f(t + \tau) = F(t) \qquad \tau = 1, 2, \ldots \tag{13.6}$$

where α is a smoothing constant between 0 and 1 chosen by the user. The best value will depend on the particular data.

Table 13.4 illustrates the exponential method, using the same data we used for the moving average. Unless we start with a historical value for $F(0)$, we cannot make a forecast for period 1. Although there are various ways to initialize the model (e.g., by averaging past observations over some interval), the choice of $F(0)$ will dissipate as time goes on. Therefore, we choose to use the simplest possible initialization method and set $F(1) = A(1) = 10$ and start the process. At time $t = 1$, our forecast for period 2 (and beyond) is $f(2) = F(1) = 10$. When we reach period 2 and observe that $A(2) = 12$, we update our smoothed estimate as follows:

$$F(2) = \alpha A(2) + (1 - \alpha)F(1) = (0.2)(12) + (1 - 0.2)(10) = 10.40$$

Our forecast for period 3 and beyond is now $f(3) = F(2) = 10.40$. We can continue in this manner to generate the remaining $f(t)$ values in Table 13.4.

Notice in Table 13.4 that when we use $\alpha = 0.6$ instead of $\alpha = 0.2$, the forecasts are much more sensitive to each new data point. For instance, in period 2, when demand increased from 10 to 12, the forecast using $\alpha = 0.2$ only increased to 10.40, while the forecast using $\alpha = 0.6$ increased to 11.20. This increased sensitivity may be good, if the model is tracking a real trend in the data, or bad, if it is overreacting to an unusual observation. Hence, analogous to our observations about the moving-average method, we can make the following points about single exponential smoothing:

1. Lower values of α will make the model more stable, but less responsive, to changes in the process being forecast.
2. The model will tend to underestimate parameters with an increasing trend, and overestimate parameters with a decreasing trend.

TABLE 13.4 Exponential Smoothing with $\alpha = 0.2$ and $\alpha = 0.6$

		Forecast $f(t)$	
Month t	Demand $A(t)$	$\alpha = 0.2$	$\alpha = 0.6$
1	10	—	—
2	12	10.00	10.00
3	12	10.40	11.20
4	11	10.72	11.68
5	15	10.78	11.27
6	14	11.62	13.51
7	18	12.10	13.80
8	22	13.28	16.32
9	18	15.02	19.73
10	28	15.62	18.69
11	33	18.09	24.28
12	31	21.08	29.51
13	31	23.06	30.40
14	37	24.65	30.76
15	40	27.12	34.50
16	33	29.69	37.80
17	50	30.36	34.92
18	45	34.28	43.97
19	55	36.43	44.59
20	60	40.14	50.83

Choosing the appropriate smoothing constant α for exponential smoothing, like choosing the appropriate value of m for the moving-average method, requires a bit of trial and error. Typically, the best we can do is to try various values of α and see which one generates forecasts that match the historical data best. For instance, Figure 13.5 plots exponential smoothing forecasts $f(t)$, using $\alpha = 0.2$ and 0.6, along with actual values $A(t)$. This plot clearly shows that the values generated using $\alpha = 0.6$ are closer to the actual data points than those generated using $\alpha = 0.2$. The increased sensitivity caused by using a high α value enabled the model to track the obvious upward trend of the data. However, because the single exponential smoothing model does not explicitly assume the existence of a trend, both sets of forecasts tended to lag behind the actual data.

Exponential Smoothing with a Linear Trend. We now turn to a model that is specifically designed to track data with upward or downward trends. For simplicity, the model assumes the trend is linear. That is, our forecasts from the present out into the future will follow a straight line. Of course, each time we receive a new observation, we will update the slope of this line, so the method can track data that change in a nonlinear fashion, although less accurately than data with a trend that is generally linear.

The basic method updates a smoothed estimate $F(t)$ and a smoothed trend $T(t)$ each time a new observation becomes available. Using these, the forecast for τ periods into the future, denoted by $f(t + \tau)$, is computed as the smoothed estimate plus τ times the smoothed trend. The equations for doing this are as follows:

$$F(t) = \alpha A(t) + (1 - \alpha)[F(t - 1) + T(t - 1)] \tag{13.7}$$

$$T(t) = \beta[F(t) - F(t - 1)] + (1 - \beta)T(t - 1) \tag{13.8}$$

$$f(t + \tau) = F(t) + \tau T(t) \tag{13.9}$$

where α and β are smoothing constants between 0 and 1 to be chosen by the user.

Notice that the equation for computing $F(t)$ is slightly different from that for exponential smoothing without a linear trend. The reason is that at period $t - 1$ the forecast for period t is given by $F(t - 1) + T(t - 1)$ (i.e., we need to add the trend for one period). Therefore, when we compute the weighted average of $A(t)$ and the current forecast, we must use $F(t - 1) + T(t - 1)$ as the current forecast.

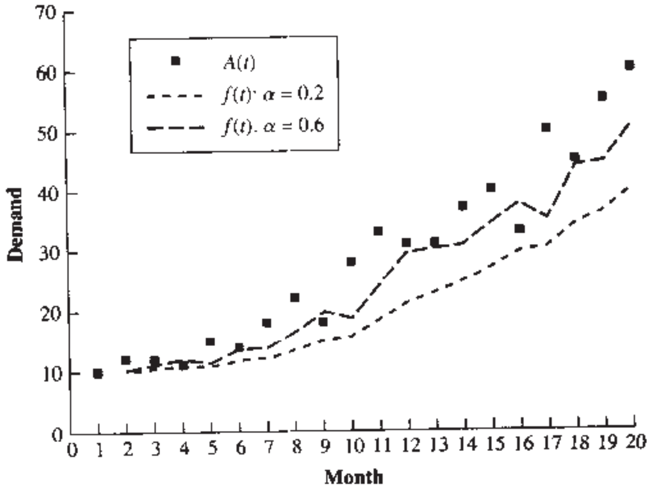


FIGURE 13.5
Exponential smoothing with $\alpha = 0.2, 0.6$

We update the trend in Equation (13.8) by computing a weighted average between the last smoothed trend $T(t-1)$ and the most recent estimate of the trend, which is computed as the difference between the two most recent smoothed estimates, or $F(t) - F(t-1)$. The $F(t) - F(t-1)$ term is like a slope. By giving this slope a weight of β (less than one), we smooth our estimate of the trend to avoid overreacting to sudden changes in the data.

As in simple exponential smoothing, we must initialize the model before we can begin. We could do this by using historical data to estimate $F(0)$ and $T(0)$. However, the simplest initialization method is to set $F(1) = A(1)$ and $T(1) = 0$. We illustrate the exponential smoothing with linear trend method using this initialization procedure, the demand data from Table 13.4, and smoothing constants $\alpha = 0.2$ and $\beta = 0.2$. For instance,

$$F(2) = \alpha A(2) + (1-\alpha)[F(1) + T(1)] = 0.2(12) + (1-0.2)(10+0) = 10.4$$

$$T(2) = \beta[F(2) - F(1)] + (1-\beta)T(1) = 0.2(10.4 - 10) + (1-0.2)(0) = 0.08$$

The remainder of the calculations are given in Table 13.5.

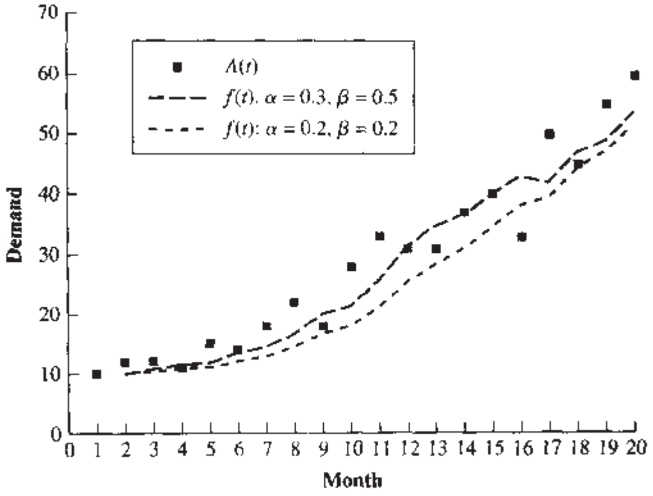
Figure 13.6 plots the forecast values $f(t)$ and the actual values $A(t)$ from Table 13.5 and plots the forecast that results from using $\alpha = 0.3$ and $\beta = 0.5$. Notice that these forecasts track these data much better than either the moving average or exponential smoothing without a linear trend. The linear trend enables this method to track the upward trend in these data quite effectively. Additionally, it appears that using smoothing coefficients $\alpha = 0.3$ and $\beta = 0.5$ results in better forecasts than using $\alpha = 0.2$ and $\beta = 0.2$. Next, we will discuss how to choose smoothing constants later in this section.

TABLE 13.5 Exponential Smoothing with a Linear Trend, $\alpha = 0.2$ and $\beta = 0.2$

Month	Demand	Smoothed Estimate	Smoothed Trend	Forecast
t	$A(t)$	$F(t)$	$T(t)$	$f(t)$
1	10	10.00	0.00	—
2	12	10.40	0.08	10.00
3	12	10.78	0.14	10.48
4	11	10.94	0.14	10.92
5	15	11.87	0.30	11.08
6	14	12.53	0.37	12.17
7	18	13.93	0.58	12.91
8	22	16.00	0.88	14.50
9	18	17.10	0.92	16.88
10	28	20.02	1.32	18.03
11	33	23.67	1.79	21.34
12	31	26.57	2.01	25.46
13	31	29.06	2.11	28.58
14	37	32.33	2.34	31.17
15	40	35.74	2.55	34.67
16	33	37.23	2.34	38.29
17	50	41.66	2.76	39.57
18	45	44.53	2.78	44.42
19	55	48.85	3.09	47.31
20	60	53.55	3.41	51.94

FIGURE 13.6

Exponential smoothing with linear trend



The Winters Method for Seasonality. Many products exhibit seasonal demand. For instance, lawn mowers, ice cream, and air conditioners have peaks associated with summer, while snow blowers, weather stripping, and furnaces have winter peaks. When this is the case, the above forecasting models will not work well because they will interpret seasonal rises in demand as permanent rises and thereby will overshoot actual demand when it declines in the off season. Likewise, they will interpret the low off-season demand as permanent and will undershoot actual demand during the peak season.

A natural way to build seasonality into a forecasting model was suggested by Winters (1960). The basic idea is to estimate a multiplicative seasonality factor $c(t)$, $t = 1, 2, \ldots,$ where $c(t)$ represents the ratio of demand during period t to the average demand during the season. Therefore, if there are N periods in the season (for example, $N = 12$ if periods are months and the season is 1 year), then the sum of the $c(t)$ factors over the season will always be equal to N. The seasonally adjusted forecast is computed by multiplying the forecast from the exponential smoothing with linear trend model (that is, $F(t) + \tau T(t)$) by the appropriate seasonality factor. The equations for doing this are as follows:

$$F(t) = \alpha \frac{A(t)}{c(t-N)} + (1-\alpha)[F(t-1) + T(t-1)] \tag{13.10}$$

$$T(t) = \beta[F(t) - F(t-1)] + (1-\beta)T(t-1) \tag{13.11}$$

$$c(t) = \gamma \frac{A(t)}{F(t)} + (1-\gamma)c(t-N) \tag{13.12}$$

$$f(t+\tau) = [F(t) + \tau T(t)]c(t) \tag{13.13}$$

where α, β, and γ are smoothing constants between 0 and 1 to be chosen by the user. Notice that Equations (13.10) and (13.11) are identical to Equations (13.7) and (13.8) for computing the smoothed estimate and smoothed trend in the exponential smoothing with linear trend model, except that the actual observation $A(t)$ is scaled by dividing by the seasonality factor $c(t-N)$. This normalizes all the observations relative to the average and hence places the smoothed estimate and trend in units of average (nonseasonal) demand. Equation (13.12) uses exponential smoothing to update the seasonality factor $c(t)$ as a weighted average of this season's ratio of actual demand to smoothed estimate

Table 13.6 The Winters Method for Forecasting with Seasonality

Year	Month	Time Period t	Actual Demand $A(t)$	Smoothed Estimate $F(t)$	Smoothed Trend $T(t)$	Seasonal Factor $c(t)$	Forecast $f(t)$
1997	Jan	1	4	—	—	0.480	
	Feb	2	2	-	—	0.240	
	Mar	3	5	—	—	0.600	
	Apr	4	8	—	—	0.960	
	May	5	11	—	—	1.320	
	Jun	6	13	—	—	1.560	
	Jul	7	18	—	—	2.160	
	Aug	8	15	—	—	1.800	
	Sep	9	9	—	—	1.080	
	Oct	10	6	—	—	0.720	
	Nov	11	5		—	0.600	
	Dec	12	4	8.33	0.00	0.480	
1998	Jan	13	5	8.54	0.02	0.491	4.00
	Feb	14	4	9.37	0.10	0.259	2.06
	Mar	15	7	9.69	0.12	0.612	5.68
	Apr	16	7	9.57	0.10	0.937	9.43
	May	17	15	9.83	0.12	1.341	12.76
	Jun	18	17	10.04	0.13	1.573	15.52
	Jul	19	24	10.26	0.13	2.178	21.97
	Aug	20	18	10.36	0.13	1.794	18.72
	Sep	21	12	10.55	0.14	1.086	11.33
	Oct	22	7	10.59	0.13	0.714	7.69
	Nov	23	8	10.98	0.15	0.613	6.43
	Dec	24	6	11.27	0.17	0.485	5.34

$A(t)/F(t)$ and last season's factor $c(t-N)$. To make the forecast in seasonal units, we multiply the nonseasonal forecast $F(t)+\tau T(t)$ by the seasonality factor $c(t)$.

We illustrate the Winters method with the example in Table 13.6. To initialize the procedure, we require a full season of seasonality factors plus an initial smoothed estimate and smoothed trend. The simplest way to do this is to use the first season of data to compute these initial parameters and then use the above equations to update them with additional seasons of data. Specifically, we simply set the smoothed estimate to be the average of the first seasons of data

$$F(N)=\frac{\sum_{t=1}^{N}A(t)}{N} \tag{13.14}$$

So, in our example, we can compute the smoothed estimate as of December 1998 to be

$$F(12)=\frac{\sum_{t=1}^{12}A(t)}{12}=\frac{4+2+\cdots+4}{12}=8.33$$

Since we are starting with only a single season of data, we have no basis for estimating a trend, so we will assume initially that the trend is zero, so that $T(N)=T(12)=0$. The model will quickly update the trend as seasons are added.[1] Finally, we compute

[1] Alternatively, one could use multiple seasons of data to initialize the model and estimate the trend from these (see Silver, Pyke, and Peterson 1998 for a method).

initial seasonality factors as the ratio of actual demand to average demand during the first season:

$$c(i) = \frac{A(i)}{\sum_{t=1}^{N} A(t)/N} = \frac{A(i)}{F(N)} \tag{13.15}$$

For instance, in our example, the initial seasonality factor for January is

$$c(1) = \frac{A(1)}{F(12)} = \frac{4}{8.33} = 0.480$$

Once we have computed values for $F(N)$, $T(N)$, and $c(1), \ldots, c(N)$, we can begin the smoothing procedure. The smoothed estimate for January 1998 is computed as

$$\begin{aligned} F(13) &= \alpha \frac{A(13)}{c(13-12)} + (1-\alpha)[F(12) + T(12)] \\ &= 0.1\left(\frac{5}{0.480}\right) + (1-0.1)(8.33+0) = 8.54 \end{aligned}$$

The smoothed trend is

$$T(13) = \beta[F(13) - F(12)] + (1-\beta)T(12) = 0.1(8.54 - 8.33) + (1-0.1)(0) = 0.02$$

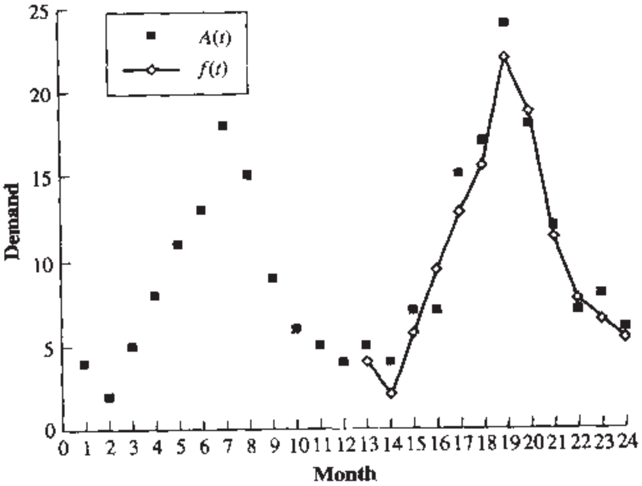
The updated seasonality factor for January is

$$c(13) = \gamma \frac{A(13)}{F(13)} + (1-\gamma)c(1) = 0.1\left(\frac{5}{8.54}\right) + (1-0.1)(0.48) = 0.491$$

The computations continue in this manner, resulting in the numbers shown in Table 13.6. We plot the actual and forecasted demand in Figure 13.7. In this example, the Winters method works very well. The primary reason is that the seasonal spike in 1998 had a similar shape to that in 1997. That is, the proportion of total annual demand that occurred in a given month, such as July, is fairly constant across years. Hence, the seasonality factors provide a good fit to the seasonal behavior. The fact that total annual demand is growing, which is accounted for by the positive trend in the model, results in an appropriately amplified seasonal spike in the second year. In general, the Winters method gives reasonable performance for seasonal forecasting where the shape of the seasonality does not vary too much from season to season.

FIGURE 13.7

The Winters method, $\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.1$



Adjusting Forecasting Parameters. All the time series models discussed involve adjustable coefficients (for example, m in the moving-average model and α in the exponential smoothing model), which must be "tuned" to the data to yield a suitable forecasting model. Indeed, we saw in Figure 13.6 that adjusting the smoothing coefficients can substantially affect the accuracy of a forecasting model. We now turn to the question of how to find good coefficients for a given forecasting situation.

The first step in developing a forecasting model is to plot the data. This will help us decide whether the data appear predictable at all, whether a trend seems to be present, or whether seasonality seems to be a factor. Once we have chosen a model, we can plot the forecast versus actual past data for various sets of parameters to see how the model behaves. However, to find a good set of coefficients, it is helpful to be more precise about measuring model accuracy.

The three most common quantitative measures for evaluating forecasting models are the *mean absolute deviation* (MAD), *mean square deviation* (MSD), and *bias* (BIAS). Each of these takes the differences between the forecast and actual values, $f(t) - A(t)$, and computes a numerical score. The specific formulas for these are as follows:

$$\text{MAD} = \frac{\sum_{t=1}^{n} |f(t) - A(t)|}{n} \tag{13.16}$$

$$\text{MSD} = \frac{\sum_{t=1}^{n} [f(t) - A(t)]^2}{n} \tag{13.17}$$

$$\text{BIAS} = \frac{\sum_{t=1}^{n} f(t) - A(t)}{n} \tag{13.18}$$

Both MAD and MSD can only be positive, so the objective is to find model coefficients that make them as small as possible. BIAS can be positive, indicating that the forecast tends to overestimate the actual data, or negative, indicating that the forecast tends to underestimate the actual data. The objective, then, is to find coefficients that make BIAS close to zero. However, note that zero BIAS does not mean that the forecast is accurate, only that the errors tend to be balanced high and low. Hence, one would never use BIAS alone to evaluate a forecasting model.

To illustrate how these measures might be used to select model coefficients, let us return to the exponential smoothing with linear trend model as applied to the demand data in Table 13.5. Table 13.7 reports the values of MAD, MSD, and BIAS for various combinations of α and β. From this table, it appears that the combination $\alpha = 0.3$, $\beta = 0.5$ works well with regard to minimizing MAD and MSD, but that $\alpha = 0.6$, $\beta = 0.6$ is better with regard to minimizing BIAS. In general, it is unlikely that any set of coefficients will be best with regard to all three measures of effectiveness. In this specific case, as can be seen in Figure 13.6, the actual data not only have an upward trend, but also tend to increase according to a nonlinear curve (i.e., the curve has a sort of parabolic shape). This nonlinear shape causes the model with a linear trend to lag slightly behind the data, resulting in a negative BIAS. Higher values of α and β give the new observations more weight and thereby cause the model to track this upward swing more rapidly. This reduces BIAS. However, they also cause it to overshoot the occasional downward dip in the data, increasing MAD and MSD.

Table 13.7 shows that the model with $\alpha = 0.3$, $\beta = 0.5$ has significantly smaller MSD than the model with our original choice of $\alpha = 0.2$, $\beta = 0.2$. This means that it fits the past data more closely, as illustrated in Figure 13.6. Since our basic assumption in using a time series forecasting model is that future data will behave similarly to past data, we should set the coefficients to provide a good fit to past data and then use these for future forecasting purposes.

Table 13.7 Exponential Smoothing with Linear Trend for Various α and β

α	β	MAD	MSD	BIAS	α	β	MAD	MSD	BIAS
0.1	0.1	10.23	146.94	−10.23	0.4	0.1	4.30	30.14	−3.45
0.1	0.2	8.27	95.31	−8.27	0.4	0.2	3.89	23.78	−2.34
0.1	0.3	6.83	64.91	−6.69	0.4	0.3	3.77	22.25	−1.77
0.1	0.4	5.83	47.17	−5.43	0.4	0.4	3.75	22.11	−1.46
0.1	0.5	5.16	36.88	−4.42	0.4	0.5	3.76	22.36	−1.29
0.1	0.6	4.69	30.91	−3.62	0.4	0.6	3.79	22.67	−1.18
0.2	0.1	6.48	60.55	−6.29	0.5	0.1	4.13	27.40	−2.84
0.2	0.2	5.04	37.04	−4.49	0.5	0.2	3.91	23.61	−1.94
0.2	0.3	4.26	27.56	−3.29	0.5	0.3	3.88	23.02	−1.49
0.2	0.4	3.90	23.75	−2.51	0.5	0.4	3.90	23.26	−1.25
0.2	0.5	3.73	22.32	−2.02	0.5	0.5	3.94	23.73	−1.10
0.2	0.6	3.65	21.94	−1.71	0.5	0.6	3.97	24.27	−1.00
0.3	0.1	4.98	37.81	−4.45	0.6	0.1	4.12	26.85	−2.42
0.3	0.2	4.11	26.30	−3.03	0.6	0.2	4.03	24.63	−1.66
0.3	0.3	3.82	22.74	−2.23	0.6	0.3	4.04	24.69	−1.29
0.3	0.4	3.66	21.81	−1.77	0.6	0.4	4.09	25.35	−1.08
0.3	0.5	3.65	21.78	−1.52	0.6	0.5	4.14	26.25	−0.95
0.3	0.6	3.68	22.06	−1.38	0.6	0.6	4.21	27.29	−0.84

The enumeration offered in Table 13.7 is given here to illustrate the impact of changing smoothing coefficients. However, in practice we do not have to use a trial-and-error approach to search for a good set of smoothing coefficients. Instead, we can use the internal optimization tool, Solver, that is included in Excel to do the search for us (see Chapter 16 for details on Solver). If we set up Solver to search for the values of α and β that (1) are between zero and one and (2) minimize MSD in the previous example, we obtain the solution $\alpha = 0.284$, $\beta = 0.467$, which attains an MSD value of 21.73. This is slightly better than the $\alpha = 0.3$, $\beta = 0.5$ solution we obtained by brute-force searching, and much faster to obtain.

Notice that in our discussion of choosing smoothing coefficients we have compared the forecast for one period into the future (i.e., the lag-1 forecast) with the actual value. However, in practice, we frequently need to forecast further into the future. For instance, if we are using a demand forecast to determine how much raw material to procure, we may need to forecast several months into the future (e.g., we may require the lag-τ forecast). When this is the case, we should use the formulas to compute the forecast for τ periods from now $f(t+\tau)$ and compare this to the actual value $A(t+\tau)$ when it occurs. The model parameters should be therefore chosen with the goal of minimizing the deviations between $f(t+\tau)$ and $A(t+\tau)$, and MAD, MSD, and BIAS should be defined accordingly.

13.3.3 The Art of Forecasting

The regression model for causal forecasting and the four time series models are representative of the vast number of quantitative tools available to assist the forecasting function. Many others exist (see Box and Jenkins (1970) for an overview of more sophisticated time series models). Clearly, forecasting is an area in which quantitative models can be of great value.

However, forecasting is more than a matter of selecting a model and tinkering with its parameters to make it as effective as possible. No model can incorporate all factors that could be relevant in anticipating the future. Therefore, in any forecasting environment, situations will arise in which the forecaster must override the quantitative model with qualitative information. For instance, if there is reason to expect an impending jump in demand (e.g., because a competitor's plant is scheduled to shut down), the forecaster may need to augment the quantitative model with this information. Although there is no substitute for experience and insight, it is a good idea to occasionally look back at past forecasting experience to see what information could have been used to improve the forecast. We won't be able to predict the future precisely, but we may be able to avoid some future blunders.

13.4 Planning for Pull

A logical and customary way to break the **production planning and control (PPC)** problem into manageable pieces is to construct a hierarchical planning framework. We illustrated a typical MRP II hierarchy in Figure 3.2. However, that framework was based on the basic MRP *push* job release mechanism. As we saw in our discussion of JIT in Chapter 4 and our comparison of push and pull in Chapter 10, pull systems offer many potential benefits over push systems. Briefly, pull systems are

1. **More efficient,** in that they can attain the same throughput as a push system with less average WIP.
2. **Easier to control,** since they rely on setting (easily observable) WIP levels, rather than release rates as do push systems.
3. **More robust,** since the performance of a pull system is degraded much less by an error in WIP level than is a push system by a comparable percentage of error in release rate.
4. **More supportive of improving quality,** since low WIP levels in pull systems both require high quality (to prevent disruptions) and facilitate it (by shortening queues and quickening detection of defects).

These benefits urge us to incorporate aspects of pull into our manufacturing control systems. Unfortunately, from a planning perspective, there is a drawback to pull. Pull systems are inherently *rate-driven*, in that we fix the level of WIP and let them run. Capacity buffers (e.g., preventive maintenance periods available to be used for overtime between shifts) are used to facilitate a very steady pace, which in turn requires highly stable demand. To achieve this, the JIT literature places considerable emphasis on production smoothing.

While a rate-driven system is logistically appealing, it is not necessarily well suited to planning. There is no natural link to customer due dates in a pull system. Customers "pull" what they need, and signals (cards or whatever) trigger replenishments. But until the demands actually occur, the system offers us no information about them. Hence, a pull system provides no inherent mechanism for planning raw material procurement, staffing, opportunities for machine maintenance, etc.

In contrast, as we noted in Chapter 5, push systems can be logistical nightmares, but are extremely well suited to planning. There is a simple and direct link between customer due dates and order releases in a push system. For instance, in a lot-for-lot MRP system, the planned order releases *are* the customer requirements (only time-phased according

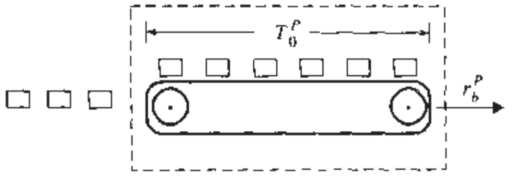


Figure 13.8

The conveyor model of a production line

to production lead times). If only the infinite-capacity assumption of MRP did not make these lead times largely fictional, we could use them to drive all sorts of planning modules. Indeed, this is precisely what MRP II systems do.

The question then is, Can we obtain the logistical benefits of pull and still develop a coherent planning structure? We think the answer is yes. But the mechanism for linking a rate-based pull system with due dates is necessarily more complex than the simple time phasing of MRP. The simplest link we know of is the **conveyor model** of a pull production line or facility, depicted in Figure 13.8 and upon which we will rely extensively in subsequent chapters.

The conveyor model is based on the observation that a pull system maintains a fairly steady WIP level, so the speed of the line and the time to traverse it are relatively constant over time. This allows us to characterize a production line with two parameters: the **practical production rate** r_b^P and the **minimum practical lead time** T_0^P. These serve the same functions as, but are somewhat different from, the **bottleneck rate** r_b and the **raw process time** T_0 of the line as defined in Chapter 7, and their ideal realizations r_b^* and T_0^* introduced in Chapter 9. Unlike the bottleneck rate, the practical production rate is the *anticipated* throughput of the line. This rate can also be standardized according to part complexity (e.g., we could count parts in units of hours of work at a bottleneck process). Thus, since r_b is the capacity of the line, we expect $r_b^P < r_b$ with utilization $u = r_b^P / r_b$. Likewise, T_0^P is the practical minimum (i.e., no queueing) *practical* time to traverse the line. This will include detractors for short-term disruptions, such as setups and routine machine failures along with routine waiting to move and any other delays that do not involve queueing. Consequently, $T_0^P > T_0$.

Using Little's law, we see that the CONWIP level W must be

$$W = r_b^P \times T_0^P$$

We can now use the conveyor model to predict when jobs will be completed by a line or process center. For instance, suppose we release a job into the line when there are already n jobs waiting in queue to be admitted into the CONWIP line (i.e., waiting for a space on the conveyor). The time until the job will be completed ℓ is given by

$$\ell = \frac{n}{r_b^P} + T_0^P = \frac{n + W}{r_b^P} \tag{13.19}$$

For example, suppose the conveyor depicted in Figure 13.8 represents a circuit board assembly line. The line runs at an average rate of $r_b^P = 2$ jobs per hour, where a job consists of a standard-size container of circuit boards. Once started, a job takes an average of $T_0^P = 8$ hours to finish. A new job that finds $n = 3$ jobs waiting to released into the line (i.e., waiting for CONWIP authorization signals) will be completed in

$$\ell = \frac{n}{r_P} + T_0^P = \frac{3}{2} + 8 = 9.5 \text{ hours}$$

on average. We revisit this problem, adding variability to the production rate in Chapter 15 where we further refine the conveyor model.

Being able to estimate output times of specific jobs allows us to address a host of planning problems:

1. If sales personnel have a means of keeping track of factory loading, they could use the conveyor model to predict how long new orders will require to fill and therefore will be able to quote reasonable due dates to customers.
2. If we keep track of how the system will evolve (i.e., what jobs will be in the line and what jobs will be waiting in queue) over time, we can "simulate" the performance of a line. This would provide the basis for a "what if" tool for analyzing the effects of different priority rules or capacity decisions on outputs. As we noted in Chapter 3, capacity requirements planning (CRP) attempts such an analysis. However, as we pointed out there, CRP uses an infinite-capacity model that invalidates predictions beyond any point where a resource becomes fully loaded. More sophisticated, finite-capacity models for making such predictions have begun to appear on the market. While more accurate than CRP, finite models frequently have massive data needs and complex computations akin to those used in discrete event simulation models. The conveyor model can simplify both data requirements and computation, as we will discuss in various contexts throughout Part III.
3. We can use the conveyor model to determine whether completions will satisfy customer due dates to develop an optimization model for setting job release times. We will do this in Chapter 15 to generate a finite-capacity scheduling tool.

By addressing these and other problems, the conveyor model can provide the linchpin of a planning framework for pull production systems. Where lines are simple enough to invoke it directly, it can be a powerful integrating tool. We give an outline of a framework that can exploit this integration. We will fill in the details and discuss generalizations to situations in which the conveyor model is overly simplistic in the remainder of Part III.

13.5 Hierarchical Production Planning

With the conveyor model to predict job completions, we can develop a hierarchical **production planning and control (PPC)** framework for pull production systems. Figure 13.9 illustrates such a hierarchy, spanning from long-term strategic issues at the top levels to short-term control issues at the bottom levels.

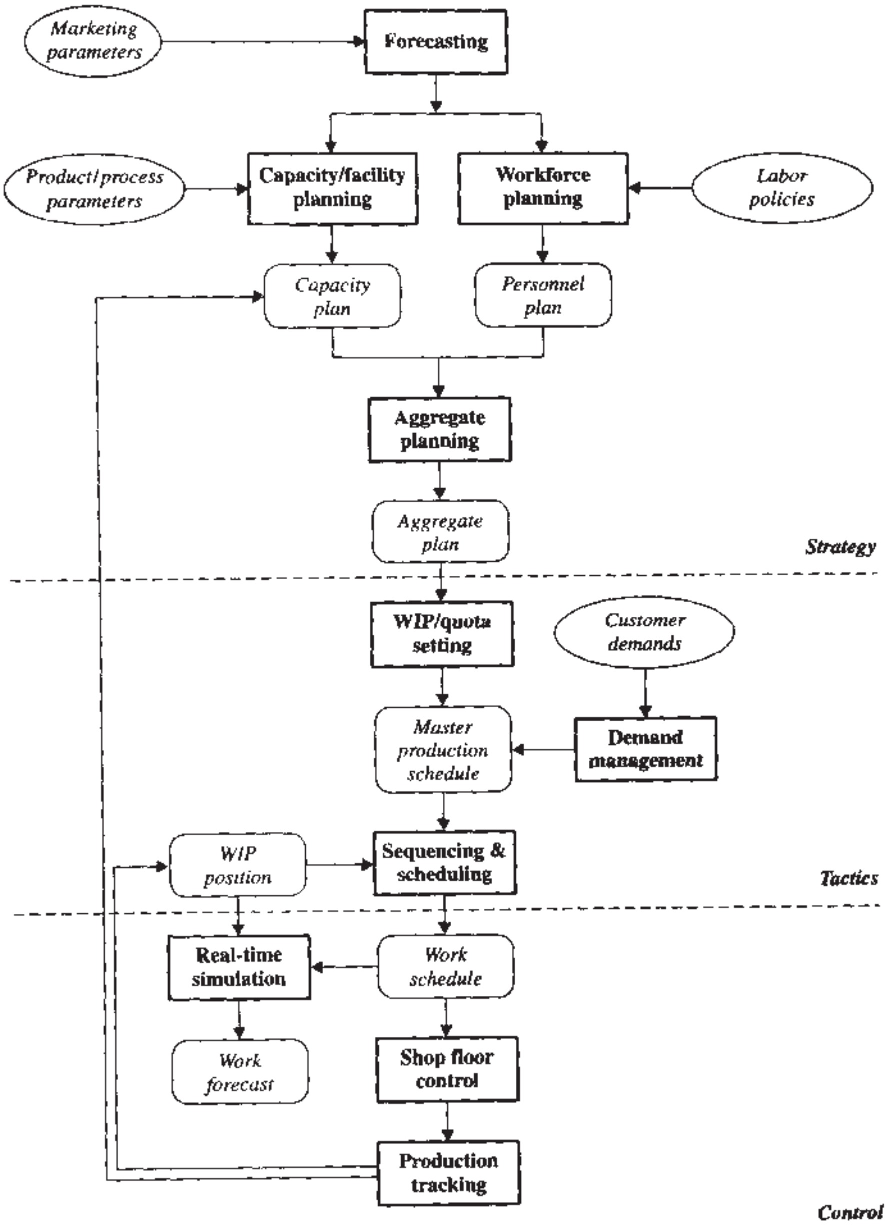
Each rectangular box in Figure 13.9 represents a separate decision problem and hence a **planning module.**[2] The rounded rectangular boxes represent *outputs* from modules, many of which are used as *inputs* in other modules. The oval boxes represent inputs to modules that are generated outside this planning hierarchy (e.g., by marketing or engineering design). Finally, the arrows indicate the *interdependence* of the modules.

The PPC hierarchy is divided into three basic levels, corresponding to long-term (strategy), intermediate-term (tactics), and short-term (control) planning. Of course, from a corporate perspective, there are levels above those shown in Figure 13.9, such as product development and business planning. Certainly these are important business

[2] We use the term *module* to represent the combination of analytic models, computer tools, and human judgment used to address the individual planning problems. As such, they are never fully automated, nor should they be.

FIGURE 13.9

A production planning and control hierarchy for pull systems



strategy decisions, and their interaction with the manufacturing function deserves serious consideration. Indeed, it is our hope that readers whose careers take them outside of manufacturing will actively pursue opportunities for greater integration of manufacturing issues into these areas. However, we will adhere to our focus on operations and assume that business strategy decisions, such as what business to be in and the nature of the product designs, have already been made. Therefore, when we speak of strategy, we are referring to *plant strategy*, which is only part of an overall business strategy.

The basic function of the long-term strategic planning tools shown in Figure 13.9 is to establish a production environment capable of meeting the plant's overall goals.

At the plant level, this begins with a **forecasting** module that takes marketing information and generates a forecast of future demand, possibly using a quantitative model like those we discussed previously. A **capacity/facility planning** module uses these demand forecasts, along with descriptions of process requirements for making the various products, to determine the needs for physical equipment. Analogously, a **workforce planning** module uses demand forecasts to generate a personnel plan for hiring, firing, training, etc., in accordance with company labor policies. Using the demand forecast, the capacity/facilities plan, and the labor plan, along with various economic parameters (material costs, wages, vendoring costs, etc.), the **aggregate planning** module makes rough predictions about future production mix and volume. The aggregate plan can also address other related issues, such as which parts to make in-house and which to contract out to external suppliers, and whether adjustments are needed in the personnel plan.

The intermediate tactical tools in Figure 13.9 take the long-range plans from the strategic level, along with information about customer orders, to generate a general plan of action that will help the plant prepare for upcoming production (by procuring materials, lining up subcontractors, etc.). A **WIP/quota-setting** module works to translate the aggregate plan into card counts and periodic production quotas required by a pull system. The production quotas form part of the **master production schedule** (MPS), which is based on the forecast demands as processed by the aggregate planning module. The MPS also contains firm customer orders, which are suitably smoothed for use in a pull production system by the *demand management* module. The **sequencing and scheduling** module translates the MPS into a work schedule that dictates what is to be worked on in the near term, for example, the next week, day, or shift.

The low-level tools in Figure 13.9 directly control the plant. The **shop floor control** module controls the real-time flow of material through the plant in accordance with this schedule, while the **production tracking** module measures actual progress against the schedule. In Figure 13.9, the production tracking module is also shown as serving a second useful function, that of feeding back information (e.g., capacity data) for use by other planning modules. Finally, the PPC hierarchy includes a **real-time simulation** module, which allows examination of what-if scenarios, such as what will happen if certain jobs are made "hot."

In the following sections, we discuss in overview fashion the issues involved at each level and the integrative philosophy for this PPC hierarchy. In this discussion, we will proceed top-down, since this helps highlight the interactions between levels. In subsequent chapters, we will provide details of how to construct the individual modules. There we will proceed bottom-up, in order to emphasize the relationship of each planning problem to the actual production process.

13.5.1 Capacity/Facility Planning

Once we have a forecast of future demand, and have made the strategic decision to attempt to fill it, we must ensure that we have adequate physical capacity. This is the function of the **capacity/facility planning** module depicted in Figure 13.9. The basic decisions to be made regarding capacity concern how much and what kind of equipment to purchase. Naturally, this includes the actual machines used to make components and final products. But it also extends to other facility issues related to the support of these machines, such as factory floor space, power supplies, air/water/chemical supplies, spare-parts inventories, material handling systems, WIP and FGI storage, and staffing levels.

Issues'that can be considered in the capacity/facility planning process include the following:

1. **Product lifetimes.** The decisions of what type and how much capacity to install depend on how long we anticipate making the product. In recent years, product lifetimes have become significantly shorter, to the point where they are frequently shorter than the physical life of the equipment. This means that the equipment must either pay for itself during the product lifetime or be sufficiently flexible to be used to manufacture other future products. Because it is often difficult to predict with any degree of confidence what future products will be, quantifying the benefits of flexibility is not easy. But it can be one of the most important aspects of facility planning, since a flexible plant that can be swiftly "tooled up" to produce new products can be a potent strategic weapon.

2. **Vendoring options.** Before the characterization of the nature of the equipment to install, a "make or buy" decision must be made, for the finished product and its subcomponents. While this is a complex issue that we cannot hope to cover comprehensively here, we offer some observations.

a. This make-or-buy decision should not be made on cost alone. Outsourcing a product because it appears that the unit cost of the vendor is lower than the (fully loaded) unit cost of making it in-house can be risky. Because unit costs depend strongly on the manner in which overhead allocation is done, a decision that seems locally rational may be globally disastrous. For example, a product that is outsourced because its unit cost is higher than the price offered by an outside supplier may not eliminate many of the overhead costs that were factored into its unit cost. Hence, these costs must be spread over the remaining products manufactured in-house, causing their unit costs to increase and making them more attractive candidates for outsourcing. There are examples of firms that have fallen into a virtual "death spiral" of repeated rounds of outsourcing on the basis of unit cost comparisons. In addition to the economic issues associated with outsourcing, there are other benefits to in-house production, such as learning effects, the ability to control one's own destiny, and tighter control over the scheduling process, that are not captured by a simple cost comparison.

b. Consideration should be given to the long term in make-or-buy decisions. We have seen companies evolve from manufacturing into distribution/service through a sequence of outsourcing decisions. While this is not necessarily a bad transition, it is certainly one that should not be made without a full awareness of the consequences and careful consideration of the viability of the firm in the marketplace as a nonmanufacturing entity.

c. When the make-or-buy decision concerns whether or not to make the product at all, then it is clearly a capacity planning decision. However, many manufacturing managers find it attractive to vendor a portion of the volume of certain products they have the capability to make in-house. Such vendoring can augment capacity and smooth the load on the plant. Since the decision of which products and how much volume to vendor depends on capacity *and* planned production, this is a question that spills over into the aggregate planning module, in which long-term production planning is done. We will discuss this problem in greater detail later and in Chapter 16. From a high-level strategic perspective, it is important to remember that giving business to outside vendors enables them to breed capabilities that may make them into competitors some day. We offer the example of IBM using Microsoft to supply the operating system for its personal computers as one example of what can happen.

3. **Pricing.** We have tried to ignore pricing as much as possible in this book, since it is a factor over which plants generally have little influence. However, in capacity decisions, a valid economic analysis simply cannot be done without some sort of forecast

of prices. We need to know how much revenue will be generated by sales in order to determine whether a particular equipment configuration is economically justified. Because prices are frequently subject to great uncertainty, this is an area in which sensitivity analysis is critical.

4. **Time value of money.** Typically, capacity increases and equipment improvements are made as capital requisitions and then depreciated over time. Interest rate and depreciation schedule, therefore, can have a significant impact on the choice of equipment.

5. **Reliability and maintainability.** As we discussed in Part II, reliability (e.g., mean time to failure (MTTF)) and maintainability (e.g., mean time to repair (MTTR)) are important determinants of capacity. Recall that availability A (the fraction of time a machine is working) is given by

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$

Obviously, all things being equal, we want MTTF to be big and MTTR to be small. But all things are never equal, as we point out in the next two observations.

6. **Bottleneck effects.** As should be clear from the discussions in Part II, capacity increases at bottleneck resources typically have a much larger effect on throughput than increases at nonbottleneck resources. Thus, it would seem that paying extra for high-speed or high-availability machines is likely to be most attractive at a bottleneck resource. However, aside from the fact that a stable, distinct bottleneck may not exist, there are problems with this overly simple reasoning, as we point out in the next observation.

7. **Congestion effects.** The single most neglected factor in capacity analysis, as it is practiced in American industry today, is variability. As we saw again and again in Part II, *variability degrades performance*. The variability of machines, which is substantially affected by failures, is an important determinant of throughput. When variability is considered, reliability and maintainability can become important factors at nonbottleneck resources as well as at the bottleneck.

We will discuss the capacity/facility analysis problem in greater detail in Chapter 18. For now, we point out that it should be done with an eye toward long-term strategic concerns and should explicitly consider variability at some level. In terms of our hierarchical planning structure, the output of a capacity planning exercise is a forecast of the physical capacity of the plant over a horizon at least long enough for the purposes of aggregate planning—typically on the order of two years.

13.5.2 Workforce Planning

As the capacity/facility planning module in Figure 13.9 determines what equipment is needed, the **workforce planning** module analogously determines what workforce is needed to support production. Both planning problems involve long-term issues, since neither the physical plant nor the workforce can be radically adjusted in the near term. So both planning modules work with long-range forecasts of demand and try to construct an environment that can achieve the system's goals. Of course, the actual sequence of events never matches the plan exactly, so both long-term capacity/facility and workforce plans are subject to short-term modification over time.

The basic workforce issues to be addressed over the long term concern how much and what kind of labor to make available. These questions must be answered within the constraints imposed by corporate labor policies. For instance, in plants with unionized labor, labor contracts may restrict who can be hired or laid off, what tasks different labor classifications can be assigned, and what hours people can work. Usually, management spends far more time hammering out the details of such agreements with labor than with determining what labor is required to support a long-term production plan. Although careful use of the workforce planning module cannot undo years of management-labor conflict, it can help both sides focus on issues that are of strategic importance to the firm.

At the root of most long-term workforce planning is a set of estimates of **the standard hours** of labor required by the products made by the plant. For example, a commercial vent hood might require 20 minutes (one-third hour) of a welder's time to assemble. If a welder is available 36 hours per week, then one welder has the capacity to produce $36 \times 3 = 108$ vent hoods per week. Thus, a production plan that calls for 540 vent hoods per week requires five welders.

Simple standard labor hour conversions can be a useful starting point for a workforce planning module. However, they fall far short of a complete representation of the issues involved in workforce planning. These issues include the following:

1. **Worker availability.** Estimates of standard labor hours must be sophisticated enough to account for breaks, vacations, training, and other factors that reduce worker availability. Many firms set "inflation factors" for converting the number of workers directly needed to the number of "onboard" workers. For instance, a multiplier of 1.4 would mean that 14 workers must be employed in order to have the equivalent of 10 directly on the jobs at all times during a given shift.

2. **Workforce stability.** Although production requirements may move up and down suddenly, it is generally neither possible nor desirable to rapidly increase and decrease the size of the workforce. A firm's ability to recruit qualified people, as well as its overall workplace attitude, can be strongly affected by changes in the size of the workforce. Some of these "softer issues" are difficult to incorporate into models but are absolutely critical to maintenance of a productive workforce.

3. **Employee training.** Training new recruits costs money *and* takes the time of current employees. In addition, inexperienced workers require time to reach full productivity. These considerations argue against sudden large increases in the workforce. However, when growth requires rapid expansion of the workforce, concerted efforts are needed to maintain the corporate culture (i.e., whatever it was that made growth occur in the first place).

4. **Short-term flexibility.** A workforce is described by more than head count. The degree of cross-training among workers is an important determinant of a plant's flexibility (its ability to respond to short-term changes in product mix and volume). Thus, workforce planning needs to look beyond the production plan to consider the unplanned contingencies (emergency customer orders, runaway success of a new product) with which the system should be able to cope.

5. **Long-term agility.** The standard labor hours approach views labor as simply another input to products, along with material and capital equipment. But workers represent more than this. In the current era, where products and processes are constantly changing, the workforce is a key source of agility (the plant's ability to rapidly reconfigure a manufacturing system for efficient production of new products as they are introduced). So-called **agile manufacturing** is largely dependent on its people, both managers and workers, to learn and evolve with change.

6. **Quality improvement.** As we noted in Chapter 12, quality, both internal and external, is the result of a number of factors, many under the direct control of workers. Educating machine operators in quality control methods, cross-training workers so that

they develop a systemwide appreciation of the quality implications of their actions, and moderating the influx of new employees so that a corporate consciousness of quality is not undermined—all these are critical parts of a plan to continuously improve quality. Although such factors are difficult to incorporate explicitly into manpower planning models, it is important that they be recognized in the overall workforce planning module.

Workforce planning is a deep and far-reaching subject that occupies a position close to the core of manufacturing management. As such, it goes well beyond operations management or factory physics. In Chapter 16 we will revisit this topic from an analytical perspective and will examine the relationship between workforce planning and aggregate planning. While this is a useful starting place for workforce planning, we remind the reader that it is only that. A well-balanced manpower plan must consider issues such as those listed previously and will require input from virtually all segments of the manufacturing organization.

13.5.3 Aggregate Planning

Once we have estimated future demand and have determined what equipment and labor will be available, we can generate an **aggregate plan** that specifies how much of each product to produce over time. This is the role of the **aggregate planning** module depicted in Figure 13.9. Because different facilities have different priorities and operating characteristics, aggregate plans will differ from plant to plant. In some facilities the dominant issue will be product mix, so aggregate planning will consist primarily of determining how much of each product to produce in each period, subject to constraints on demand, capacity, and raw material availability. In other facilities, the crucial issue will be the timing of production, so the aggregate planning module will seek to balance the costs of production (e.g., overtime and changes in the workforce size) with the costs of carrying inventory while still meeting demand targets. In still others, the focus will be on the timing of staff additions or reductions. In all these, we may also include the possibility of augmenting capacity through the use of outside vendors.

Regardless of the specific formulation of the aggregate planning problem, it is valuable to be able to identify which constraints are binding. For instance, if the aggregate planning module tells us that a particular process center is heavily utilized on average over the next year, then we know that this is a resource that will have to be carefully managed. We may want to institute special operating policies, such as using floating labor, to make sure this process keeps working during breaks and lunches. If the problem is serious enough, it may even make sense to go back and revise the capacity and manpower plans and requisition additional machinery and/or labor if possible.

The decisions that are addressed by the aggregate planning module require a fair amount of advance planning. For instance, if we are seeking to build up inventory for a period of peak demand during the summer, clearly we must consider the production plan for several months prior to the summer. If we want to consider staffing changes to accommodate the production plan, we may require even more advance warning. This generally means that the planning horizon for aggregate planning must be relatively long, typically a year or more. Of course, we should regenerate our aggregate plan more frequently than this, since a year-long plan will be highly unreliable toward the end. It often makes sense to update the aggregate plan quarterly or biannually.

We give specific formulations of representative aggregate planning modules in Chapter 16. Because we can often state the problem in terms of minimizing cost subject to meeting demand, we frequently use the tool of linear programming to help solve the aggregate planning problem. Linear programming has the advantages that

1. It is very fast, enabling us to solve large problems quickly. This is extremely important for using the aggregate planning module in what-if mode.
2. It provides powerful sensitivity analysis capability, for instance, calculating how much additional capacity would affect total cost. This enables us to identify critical resources and quickly gauge the effectiveness of various changes.

As we will see in Chapter 16, linear programming also offers us a great deal of flexibility for representing different aggregate planning situations.

13.5.4 WIP and Quota Setting

The **WIP/quota-setting** module, depicted in Figure 13.9 as working in close conjunction with the aggregate planning module, is needed to translate the aggregate plan to control parameters for a pull system. Recall that the key controls in a pull system are the WIP levels, or card counts, in the production lines. Also, to link the pull system to customer due dates, we need to set an additional control, namely, the production quota. By establishing a quota, and then using buffer capacity to ensure that the quota is met with regularity, we make the system behavior approximate that of the "conveyor model" discussed. The predictability of the conveyor model allows us to coordinate system outputs with customer due dates.

Card Counts. We include **WIP setting,** or card count setting, at the intermediate level in the PPC hierarchy in Figure 13.9, instead of at the bottom level, to remind the reader that WIP levels should not be adjusted too frequently. As we noted in Chapter 10, WIP is a fairly insensitive control. Altering card counts in an effort to cause throughput to track demand is not likely to work well because the system will not respond rapidly enough. Therefore, like other decisions at this level in the hierarchy, WIP levels should be reevaluated on a fairly infrequent basis, say, quarterly.

Fortunately, the fact that WIP is an insensitive control also makes it relatively easy to set. As long as WIP levels are adequate to attain the desired throughput and are not grossly high, the system will function well. Therefore, it does not make sense to develop highly sophisticated tools for computing WIP levels. In systems that are moving from push to pull, it probably makes sense to set the initial WIP levels in the pull system equal to the average levels that were experienced under push. Then, once the system is operating stably, make incremental reductions. If a kanban-type system is used, so that WIP levels are set at different points in the line, remove cards from those stations with long queues that never or rarely empty out. If a CONWIP system is used, then the overall WIP level can be reduced incrementally. Once workable WIP levels have been established, they should be adjusted infrequently, to ensure that changes are made in response to long-term trends rather than short-term fluctuations.

If we must set the WIP level along a *new* (or reconfigured) routing that is to be run as a CONWIP line, we cannot rely on historical performance to gauge the appropriate WIP level. In this situation, the following is a reasonable rule of thumb. First, establish a desired and *feasible* cycle time for the routing CT and identify the practical production rate r_b^P (e.g., a feasible fraction of the bottleneck rate r_b). Then use Little's law to solve for the WIP level as

$$\text{WIP} = r_b^P \times \text{CT}$$

If r_b^P and CT are realistic, this method will yield a reasonable starting point for WIP, which can be adjusted over time. In general, care must be taken not to underestimate the feasible cycle time or practical production rate, since this will result in too little WIP, with the consequence that throughput will be too low.

Production Quotas. In addition to WIP levels, the other key parameter for controlling a pull system is the production quota. Hence, **quota setting** is included with the WIP setting module in the PPC hierarchy in Figure 13.9.

The basic idea of a production quota is that we establish a periodic quantity of work that we will (almost) always complete during the quota period. The period under question might be a shift, a day, or a week. In its strictest form, a production quota means that

1. Production during the period stops when quota is reached.
2. Overtime is used at the end of the period to make up any shortage that occurred during regular time.

This allows us to count on a steady output and therefore facilitates planning and due date quoting. Of course, in practice, few quota systems adhere rigidly to this protocol. Indeed, one of the benefits of CONWIP that we cited in Chapter 10 is that it allows working ahead of the schedule when circumstances permit. However, for the purposes of planning a reasonable periodic production quota, it makes sense to model the system as if we stop when quota is reached.

Establishing an economic production quota requires consideration of both cost and capacity data. Relevant costs are those related to lost throughput and overtime. Important capacity parameters include both the mean *and* the standard deviation of production during a specified time interval (e.g., a week or a day). Standard deviation is needed because variability of output has an impact on our ability to make a given production quota. In general, the more variable the production process, the more likely we are to miss the quota.

To see this, consider Figure 13.10. Suppose we have set the production quota for regular time production (e.g., Monday through Friday) to be Q units of work.[3] If we do not make Q units during regular time, then we must run overtime (e.g., Saturday and Sunday) to make up the shortage. Because of the usual contingencies (machine failures, worker absenteeism, yield loss, etc.), the actual amount of work completed during regular time will vary from period to period. Figure 13.10 represents two possible distributions of regular time production that have the same mean μ but different standard deviations σ. The probability of missing the quota is represented by the area under each curve to the left of the value Q. Since the area under curve A, with the smaller standard deviation, is less than that under B, the probability of missing the quota is less. What this means is that if we define a probability of missing the quota that we are willing to live with—a "service level" of sorts—then we will be able to set a higher quota for curve A than for curve B. We can aim closer to capacity because the greater predictability of curve A gives us more confidence in our ability to achieve our goal with regularity.

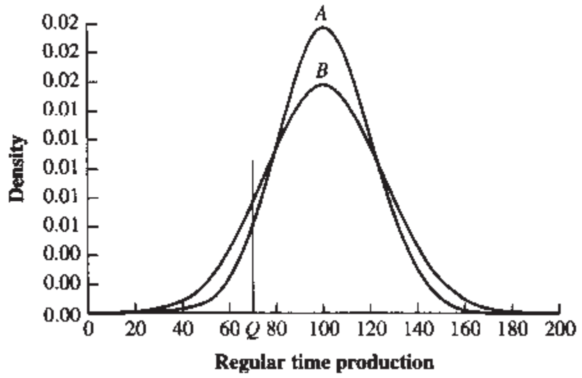
This analysis suggests that if we knew the mean μ and standard deviation σ of regular time production,[4] a very simple way to set a production quota would be to calculate the quota we can achieve S percent of the time, where S is chosen by the user. If regular time production X can be reasonably approximated by the normal distribution, then we

[3] In a simple, single-product model, units of work are equal to physical units. In a more complex, multiproduct situation, units must be adjusted for capacity, for instance, by measuring them in hours required at a critical resource.

[4] We will discuss a mechanism for obtaining estimates of μ and σ from actual operating experience in Chapter 14.

FIGURE 13.10

Probability of missing quota under different production distributions



can compute the appropriate quota by finding the value Q that satisfies

$$\Phi\left(\frac{Q-\mu}{\sigma}\right) = 1 - S$$

where $\Phi(\cdot)$ represents the cdf of the standard normal distribution.

For example, suppose that $\mu = 100$, $\sigma = 10$, and we have selected $S = 85$ percent as our service level. Then the quota Q is the value for which

$$\Phi\left(\frac{Q-100}{10}\right) = 1 - 0.85 = 0.15$$

From a standard normal table, we find that $\Phi(-1.04) = 0.15$. Therefore, we can find Q from

$$\frac{Q-100}{10} = -1.04$$
$$Q = 89.6$$

A problem with this simple method is that it considers only capacity, not costs. Therefore it offers no guidance as to whether the chosen service level is appropriate. A lower service level will result in a higher quota, which will increase throughput but will also increase overtime costs. A higher service level will result in a lower quota, which will reduce throughput and overtime costs. We offer a model for balancing the cost of lost throughput with the cost of overtime in Appendix 13A and more complex variations on this model in Hopp et al. (1993).

13.5.5 Demand Management

The effectiveness of any production control system is greatly determined by the environment in which it operates. A simple flow line can function well with very simple planning tools, while a complex job shop can be a management nightmare even with very sophisticated tools. This is just a fact of life; some plants are easier to manage than others. But it is also a good reason to remember one of our "lessons of JIT," namely, that *the environment is a control.* For example, if managers can make a job shop look like a flow shop by dedicating machines to "cells" for making particular groups of products, they can greatly simplify the planning and control process.

One key area in which we can shape the environment "seen" by the modules in the lowest levels of the planning hierarchy is in managing customer demands. The **demand management** module shown in Figure 13.9 does this by filtering and possibly adjusting customer orders into a form that produces a manageable master production schedule. As we noted in Chapter 4, leveling demand or "production smoothing" is an essential feature of JIT. Without a stable production volume and product mix, the rate-driven, mixed-model production approach described by Ohno (1988) and the other JIT advocates cannot work. This implies that customer orders cannot be released to the factory in the random order in which they are received. Rather, they must be collected and grouped in a way that maintains a fairly constant loading on the factory. Balancing the concern for factory stability with the desire for dependable customer service and short competitive due date quotes is the challenge of the demand management module.

There are many approaches one could use to quote due dates and establish a near-term MPS within the demand management module. As we discussed, if we establish periodic production quotas, then we can use the conveyor model for predicting flow through the plant. Under these conditions, we can think of customer due date quoting as "loading the conveyor." If we do not have to worry about machine setups and have a capacity cushion, we can quote due dates in the order they are received, using the conveyor model described by Equation (13.19). However, when there is variability and little or no capacity cushion, we must quote due dates using a different procedure (see Chapter 15). Likewise, if batching products according to family (i.e., parts that share important machine setups) is important to throughput, we may want to use some of the sequencing techniques discussed in Chapter 15.

While there are many methods, the important point is not *which* method but that *some* method be employed. Almost anything that achieves consistency with the scheduling procedure will be better than the all-too-common approach of quoting due dates in near isolation from the manufacturing process.

13.5.6 Sequencing and Scheduling

The MPS is still a production *plan*, which must be translated to a work schedule in order to guide what actually happens on the factory floor. In the MRP II hierarchy, shown in Figure 3.2, this figure is carried out by MRP.[5] In the production planning and control hierarchy for pull systems shown in Figure 13.9, we include a **sequencing/scheduling module** that is the pull analog of MRP. As in MRP, the objective of this sequencing/scheduling module is to provide a schedule that governs release times of work orders and materials and then facilitates their movement through the factory.

To paraphrase Einstein, we should strive to make the work schedule as simple as possible, but no simpler. The goal should be to provide people on the floor with enough information to enable them to make reasonable control choices, but not so much as to overly restrict their options or make the schedule unwieldy. What this means in practice is that different plants will require different scheduling approaches. In a simple flow line with no significant setup times, a simple sequence of orders, possibly arranged according to earliest due date (EDD), may be sufficient. Maintaining a first-in-system-first-out (FISFO) ordering of jobs at the other stations will yield a highly predictable and easily manageable output stream for this situation.

However, in a highly complex job shop, with many routings, machine setups, and assemblies of subcomponents, a simple sequence is not even well defined, let alone useful. In the more complex situations, it will not be clear that the MPS is feasible.

[5] Recall from Chapter 3 that MRP ("little mrp") refers to *material requirements planning*, the tool for generating planned order releases, while MRP II ("big MRP") refers to *manufacturing resources planning*, the overarching planning system incorporating MRP. *Enterprise resource planning (ERP)* extends the MRP II hierarchy to multiple-facility systems.

Consequently, iteration between the MPS module and the sequencing/scheduling module will be necessary. A procedure for detecting schedule infeasibility and suggesting remedies (e.g., adding capacity, pushing out due dates) is called **capacitated material requirements planning,** or **MRP-C,** and is described in Chapter 15. This procedure integrates the demand management, MPS, and sequencing/scheduling functions into one. In complex situations such as this, we may need to provide a fairly detailed schedule, with specific release times for jobs and materials and predicted arrival times of jobs at workstations. Of course, the data requirements and maintenance overhead of the system required to generate such a schedule may be substantial, but this is the price we pay for complexity.

13.5.7 Shop Floor Control

Regardless of how accurate and sophisticated the scheduling tool is, the actual work sequence never follows the schedule exactly. The **shop floor control (SFC)** module shown in Figure 13.9 uses the work schedule as a source of general guidance, adhering to it whenever possible, but also making adjustments when necessary. For instance, if a machine failure delays the arrival of parts required in an assembly operation, the SFC module must determine how the work sequence should be changed. In theory, this can be an enormously complex problem, since the number of options is immense—we could wait for the delayed part, we could jump another job ahead in the sequence, we could scramble the entire schedule, and so on. But, in practice, we must make decisions quickly, in real time, and therefore cannot hope to consider every possibility. Therefore, the SFC module must restrict attention to a reasonable class of actions and help the user make effective and robust choices.

To take advantage of the pull benefits we discussed in Chapter 10, we favor an SFC module based on a pull mechanism. The CONWIP protocol is perhaps the simplest approach and therefore deserves at least initial consideration. To use CONWIP in conjunction with the sequencing/scheduling module, we establish a WIP cap and do not allow releases into the line when the WIP exceeds the maximum level. This will serve to delay releases when the plant is behind schedule and further releases cannot help. CONWIP also provides a mechanism for working ahead of the schedule when things are going well. If the WIP level falls below the WIP cap before the next job is scheduled to be released, we may want to allow the job to start anyway. As long as we do not work too far ahead of the schedule and cause a loss of flexibility by giving parts "personality" too early, this type of work-ahead protocol can be very effective.

Chapter 14 is devoted to the SFC problem; there we will discuss implementation of CONWIP-type SFC modules and will identify situations in which more complicated SFC approaches may be necessary.

13.5.8 Real-Time Simulation

In a manufacturing management book such as this, one is tempted to make sweeping admonitions of the form "Never have hot jobs," and "Always follow the published schedule." Certainly, the factory would be easier to run if such rigid rules could be followed. But the ultimate purpose of a manufacturing plant is not to make the lives of its managers easy; it is to make money by satisfying customers. Since customers change their minds, ask for favors, etc., the reality of almost every manufacturing environment is that sometimes emergencies occur and therefore some jobs must be given special treatment. One would hope that this doesn't occur all the time (although it all too frequently does, as in a plant we once visited where *every* job shown on the MRP system had been designated "rush"). But, given that it will happen, it makes sense to design the planning system to survive these eventualities, and even provide assistance with them. This is the job of the **real-time simulation** module shown in Figure 13.9.

We have found simulation to be useful in dealing with emergency situations, such as hot jobs. By simulation, however, we do not mean full-blown Monte Carlo simulation with random number generators and statistical output analysis. Instead, we are referring to a very simple deterministic model that can mimic the behavior of the factory for short periods of time. One option for doing this is to make use of the previously described conveyor model to represent the behavior of process centers and take the current position of WIP in the system, a list of anticipated releases, and a set of capacity data (including staffing), to generate a set of job output times. Such a model can be reasonably accurate in the near term (e.g., over the next week), but because it cannot incorporate unforeseen events such as machine failures, it can become very inaccurate over the longer term. Thus, as long as we restrict the use of such a model to answering short-term what-if questions—What will happen to due date performance of various other jobs if we expedite job n?—this type of tool can be very useful. Knowing the likely consequences in advance of taking emergency actions can prevent causing serious disruption of the factory for little gain.

13.5.9 Production Tracking

In the real world there will always be contingencies that require human intervention by managers. While this may seem discouraging to the designers of production planning systems, it is one of the key reasons for the existence of manufacturing managers. A good manager should strive for a system that functions smoothly most of the time, but also be ready to take corrective action when things do not function smoothly. To detect problems in a timely fashion and formulate responses, a manager must have key data at her fingertips. These data might include the location of parts in the factory, status of equipment (e.g., up, down, under repair), and progress toward meeting schedule. The **production tracking** module depicted in Figure 13.9 is responsible for tabulating and displaying this type of data in a usable format.

Many of the planning modules in Figure 13.9 rely on estimated data. In particular, capacity data are essential to several planning decisions. A widely used practice for estimating capacity of currently installed equipment is to start with the rated capacity (e.g., in parts per hour) and reduce this number according to various detractors (machine downtime, operator unavailability, setups, etc.). Since each detractor is subject to speculation, such estimates can be seriously in error. For this reason, it makes sense to use the production tracking module to collect and update capacity data used by other planning modules. As we will see in Chapter 14, we can use the technique of exponential smoothing from forecasting to generate a smoothed estimate of capacity and to monitor trends over time.

13.6 Conclusions

In this chapter, we have offered an overview of a production planning and control hierarchy that is consistent with the pull production systems we discussed in Chapters 4 and 10. This overview was necessarily general, since there are many ways a planning system could be constructed and different environments are likely to require different systems. We will fill in specifics in subsequent chapters on the individual planning modules. For now, we close with a summary of the main points of this chapter pertaining to the overall structure of a planning hierarchy:

1. *Planning should be done hierarchically.* It makes no sense to try to use a precise, detailed model to make general, long-term decisions on the basis of rough, speculative data. In general, the shorter the planning horizon, the more details are required. For this reason, it is useful to separate planning problems into long-term (strategic), intermediate-term (tactical), and short-term (control) problems. Similarly, the level of detail about products increases with nearness in time, for instance, planning for total volume in the very long term, part families in the intermediate term, and specific part numbers in the very short term.

2. *Consistency is critical.* Good individual modules can be undermined by a lack of coordination. It is important that common capacity assumptions, consistent staffing assumptions, and coordinated data inputs be used in the different planning modules.

3. *Feedback forces consistency and learning.* Some manufacturing managers continue to use poor-quality data without checking their accuracy or setting up a system for collecting better data from actual plant performance. Regardless of how it is done (e.g., manually or in automated fashion), it is important to provide some kind of feedback for updating critical parameters. Furthermore, by providing a mechanism for observing and tracking progress, feedback promotes an environment of continual improvement.

4. *Different plants have different needs.* The above principles are general; the details of implementing them must be specific to the environment. Small, simple plants can get away with uncomplicated manual procedures for many of the planning steps. Large, complex plants may require sophisticated automated systems. Although we will be as specific as possible in the remainder of Part III, the reader is cautioned against taking details too literally; they are presented for the purposes of illustration and inspiration and cannot replace the thoughtful application of basics, intuition, and synthesis.

Appendix 13A
A Quota-Setting Model

The key economic tradeoff to consider in the quota-setting module is that between the cost of lost throughput and the cost of overtime. High production quotas tend to increase throughput, but run the risk of requiring more frequent overtime. Low quotas will reduce overtime, but will also reduce throughput.

To develop a specific quota-setting model, let us consider regular time consisting of Monday through Friday (three shifts per day) with Saturday available for preventive maintenance (PM) and catch-up. If catch-up time is needed, we assume a full shift is worked (e.g., union regulations or company policy requires it). Consequently the cost of overtime is essentially fixed, and we will represent it by C_{OT}. If we let the net profit per standardized unit of production be p and the total expected profit (net revenue minus expected overtime cost) be denoted by Z, the quota-setting problem can be formally stated as

$$\max_{Q} \; Z = pQ - C_{\text{OT}}\text{P (overtime is needed)} \tag{13.20}$$

Notice that, as expected, decreasing Q affects the objective by lost sales, while increasing Q will affect it by increasing the probability that overtime will be needed. The optimization problem is to find the value of Q that strikes the right balance.

Where shifts are long compared to the time to produce one part, it may be reasonable to assume that production during regular time is normally distributed with mean μ and standard deviation σ. This assumption allows us to express the weekly quota as $Q = \mu - k\sigma$. Now the question becomes, How many standard deviations below mean production should we set the quota to be? In other words, our decision variable is now k. Under this assumption, we can rewrite Equation (13.20) as

$$\max_k \ Z = p(\mu - k\sigma) - C_{\text{OT}}[1 - \Phi(k)] \tag{13.21}$$

where $\Phi(k)$ represents the cumulative distribution function of the standard normal distribution.

It not difficult to show (although we will not burden the reader with the details) that the unique solution to Equation (13.21) is

$$k^* = \sqrt{2 \ln \frac{C_{\text{OT}}}{\sqrt{2\pi}\, p\sigma}} \tag{13.22}$$

We can then express the optimal quota directly in units of work, instead of units of standard deviations, as follows:

$$Q^* = \mu - k^*\sigma \tag{13.23}$$

Notice that since k^* will never be negative, Equation (13.23) implies that the optimal quota will always be less than mean regular time production. As long as overtime costs are sufficiently high to make using overtime on a routine basis unattractive, this result will be reasonable. If we were to use a quota *equal* to the mean regular time production, then we would expect to miss it, and require overtime, approximately 50 percent of the time. Hence, if overtime is sufficiently expensive, less frequent use of it will be economical; therefore we should choose a quota less than the mean regular time production, and this model is plausible.

However, it is quite possible that the profitability of additional sales outweighs the cost of overtime. In this situation, our intuition tells us that a high quota (i.e., to force additional production) may be attractive, even if it results in missing the quota more than 50 percent of the time. For instance, consider an example with the following costs and production parameters:

$$p = \$100 \qquad \mu = 5{,}000$$
$$C_{\text{OT}} = \$10{,}000 \qquad \sigma = 500$$

Notice that we can "pay" for overtime with the profits of just 100 units, which is only 2 percent of the mean regular time production. This means that there is strong incentive to use the overtime period for extra production. Using our model to analyze this issue by substituting the above numbers into expression (13.22), we get

$$k^* = \sqrt{-5.06}$$

which is mathematically ridiculous. Clearly, the model runs into trouble whenever

$$\frac{C_{\text{OT}}}{\sqrt{2\pi}\, p\sigma} < 1 \tag{13.24}$$

because the natural logarithm term in Equation (13.22) becomes negative. In economic terms, this means that the fixed cost of overtime is not large enough to discourage the use of overtime for routine production. In practical terms, it means either of the following:

1. The fixed overtime cost should be reexamined, and perhaps altered. It may also make sense to include a variable (i.e., per unit) overtime cost. Development of such a model is given in Hopp, Spearman, and Duenyas (1993).
2. It may really be economically attractive to use overtime for routine production. If this is the case, it may make sense to run continuously, without capacity cushions. To set a target quota for the purposes of quoting due dates to customers, we need to balance the cost of running at less than maximum capacity with the cost of failing to meet a promised due date. A model for this case is also described in Hopp, Spearman, and Duenyas (1993).

The above simple model can be used to give a rough measure of the economics of capacity parameters. Clearly, Equations (13.21) and (13.22) indicate that both the mean and the standard deviation of regular time production are important. By using these equations, we can compute the effect on the weekly profit of changes in various parameters. In particular, we can examine the

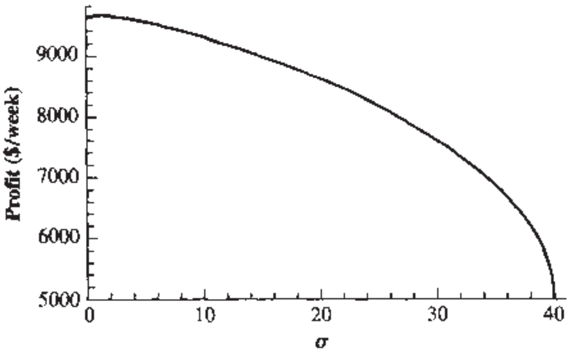


FIGURE 13.11

Weekly profit as a function of σ when μ = 100

effect of changes in the mean of regular time production μ and standard deviation of regular time production σ.

To see this, consider a simple example in which $p = \$100$, $C_{OT} = \$10{,}000$, and μ and σ are varied to determine their impact. From Equation (13.21) it is obvious that profit will increase linearly in mean regular time capacity μ. If σ is fixed, k^* does not change when μ is varied. Therefore, each increase in μ by 1 unit increases Z by p. Obviously, we are able to make more and therefore sell more.[6]

The situation is a little more complex when μ is fixed but σ is varied. This is because (from (13.22)) k^* will change as σ is altered. Furthermore, we must be careful that the term inside the square root of Equation (13.22) does not become negative. Condition (13.24) implies that we must have

$$\sigma > \frac{C_{OT}}{\sqrt{2\pi}\,p} = \frac{10,000}{\sqrt{2\pi}\,100p} = 39.9$$

for k^* to be well defined. Figure 13.11 plots the optimal weekly profit when μ is fixed at 100 units and σ is varied from 0 to 39.9. This figure illustrates the general result that profits increase when variability is reduced. The reason for this is that when regular time production is less variable, we can set quota closer to capacity without risking frequent overtime. Thus, we can achieve greater sales revenues without incurring greater overtime costs.

Study Questions

1. Why does it make sense to address the problems of planning and control in a manufacturing system with a hierarchical system? What would a nonhierarchical system look like?
2. Is it reasonable to specify rules regarding the frequency of regeneration of particular planning functions (e.g., "aggregate planning should be done quarterly")? Why or why not?
3. Give some possible reasons why MRP has spawned elaborate hierarchical planning structures while JIT has not.
4. Why is it important for the various modules in a hierarchical planning system to achieve consistency? Why is such consistency not always maintained in practice?
5. What is the difference between *causal forecasting* and *time series forecasting?*
6. Why might an exponential smoothing model exhibit negative bias? An exponential smoothing model with a linear trend?
7. In this era of rapid change and short product lifetimes, it is common for process technology to be used to produce several generations of a product or even completely new products. How might this fact enter into the decisions related to capacity/facility planning?

[6] Note that this is only true because of our assumption that capacity is the constraint on sales. If demand becomes the constraint, then this is clearly no longer true, since it makes no sense to set the quota beyond what can be sold.

8. In what ways are capacity/facility planning and workforce planning analogous? How do they differ?
9. How must the capacity/facility planning and aggregate planning be coordinated? What can happen if they are not?
10. One of the functions of sequencing and scheduling is to make effective use of capacity by balancing setups and due dates. This implies that actual capacity is not known until a schedule is developed. But both the capacity/facility planning and aggregate planning functions rely on capacity data. How can they do this in the absence of a schedule (i.e., how can they be done at a higher level in the hierarchy than sequencing or scheduling)?
11. How is demand management practiced in MRP? In JIT?
12. If a plant generates a detailed schedule at the beginning of every week, does it need a shop floor control module? If so, what functions might an SFC module serve in such a system?
13. What purpose does feedback serve in a hierarchical production planning system?

Problems

1. Suppose the monthly sales for a particular product for the past 20 months have been as follows:

Month	1	2	3	4	5	6	7	8	9	10
Sales	22	21	24	30	25	25	33	40	36	39
Month	11	12	13	14	15	16	17	18	19	20
Sales	50	55	44	48	55	47	61	58	55	60

a. Use a five-period moving average to compute forecasts of sales for months 6 to 20 and a seven-period moving average to compute forecasts for months 8 to 20. Which fits the data better for months 8 to 20? Explain.

b. Use an exponential smoothing approach with smoothing constant $\alpha = 0.2$ to forecast sales for months 2 to 20. Change α to 0.1. Does this make the fit better or worse? Explain.

c. Using exponential smoothing, find the value of α that minimizes the mean squared deviation (MSD) over months 2 to 20. Find the value of α that minimizes BIAS. Are they the same? Explain.

d. Use an exponential smoothing with a linear trend and smoothing constants $\alpha = 0.4$ and $\beta = 0.2$ to predict output for months 2 to 20. Does this fit better or worse than your answers to *b*? Explain.

2. The following data give closing values of the Dow Jones Industrial Average for the 30 weeks, months, and years prior to August 1, 1999.

a. Use exponential smoothing with a linear trend and smoothing coefficients of $\alpha = \beta = 0.1$ on each set of data to generate forecasts for the Dow Jones Industrial Average on August 1, 2000. Which data set do you think yields the best forecast?

b. What weight does a one-year-old data point get when we use smoothing constant $\alpha = 0.1$ on the weekly data? On the monthly data? On the annual data? What smoothing constant for the monthly model that gives the same weight to one-year-old data is given by the annual model with $\alpha = 0.1$?

c. Does using the adjusted smoothing constant computed in part *b* (for α and β) in the monthly model make it predict the closing price for August 1, 2000? If not, why not?

d. How much value do you think time series models have for forecasting stock prices? What features of the stock market make it difficult to predict, particularly in the short term?

Weekly Data		Monthly Data		Annual Data	
Date	**Close**	**Date**	**Close**	**Date**	**Close**
1/4/99	9,643.3	2/1/97	6,877.7	8/1/69	836.7
1/11/99	9,340.6	3/1/97	6,583.5	8/1/70	764.6
1/18/99	9,120.7	4/1/97	7,009.0	8/1/71	898.1
1/25/99	9,358.8	5/1/97	7,331.0	8/1/72	963.7
2/1/99	9,304.2	6/1/97	7,672.8	8/1/73	887.6
2/8/99	9,274.9	7/1/97	8,222.6	8/1/74	678.6
2/15/99	9,340.0	8/1/97	7,622.4	8/1/75	835.3
2/22/99	9,306.6	9/1/97	7,945.3	8/1/76	973.7
3/1/99	9,736.1	10/1/97	7,442.1	8/1/77	861.5
3/8/99	9,876.4	11/1/97	7,823.1	8/1/78	876.8
3/15/99	9,903.6	12/1/97	7,908.3	8/1/79	887.6
3/22/99	9,822.2	1/1/98	7,906.5	8/1/80	932.6
3/29/99	9,832.5	2/1/98	8,545.7	8/1/81	881.5
4/5/99	10,173.8	3/1/98	8,799.8	8/1/82	901.3
4/12/99	10,493.9	4/1/98	9,063.4	8/1/83	1,216.2
4/19/99	10,689.7	5/1/98	8,900.0	8/1/84	1,224.4
4/26/99	10,789.0	6/1/98	8,952.0	8/1/85	1,334.0
5/3/99	11,031.6	7/1/98	8,883.3	8/1/86	1,898.3
5/10/99	10,913.3	8/1/98	7,539.1	8/1/87	2,663.0
5/17/99	10,829.3	9/1/98	7,842.6	8/1/88	2,031.7
5/24/99	10,559.7	10/1/98	8,592.1	8/1/89	2,737.3
5/31/99	10,799.8	11/1/98	9,116.6	8/1/90	2,614.4
6/7/99	10,490.5	12/1/98	9,181.4	8/1/91	3,043.6
6/14/99	10,855.6	1/1/99	9,358.8	8/1/92	3,257.4
6/21/99	10,552.6	2/1/99	9,306.6	8/1/93	3,651.3
6/28/99	11,139.2	3/1/99	9,786.2	8/1/94	3,913.4
7/5/99	11,193.7	4/1/99	10,789.0	8/1/95	4,610.6
7/12/99	11,209.8	5/1/99	10,559.7	8/1/96	5,616.2
7/19/99	10,911.0	6/1/99	10,970.8	8/1/97	7,622.4
7/26/99	10,655.1	7/1/99	10,655.1	8/1/98	7,539.1
8/2/99	10,714.0	8/1/99	10,829.3	8/1/99	10,829.3

3. Hamburger Heaven has hired a team of students from the local university to develop a forecasting tool for predicting weekly burger sales to assist in the purchasing of supplies. The assistant manager, who has taken a couple of college classes, has heard of exponential smoothing and suggests that the students try using it. He gives them the following data on sales for the past 16 weeks.

Week	1	2	3	4	5	6	7	8
Sales	3,500	3,700	3,400	3,900	4,100	3,500	3,600	4,200
Week	9	10	11	12	13	14	15	16
Sales	9,300	8,900	9,100	9,200	9,300	9,000	9,400	9,100

a. What happens if exponential smoothing (with no trend) is applied to these data in a conventional manner? Use a smoothing constant $\alpha = 0.3$.

b. Does it improve the forecast if we use exponential smoothing with a linear trend and smoothing constants $\alpha = \beta = 0.3$?

c. Suggest a modification of exponential smoothing that might make more sense for this situation.

4. Select-a-Model offers computer-generated photos of people posing with famous supermodels. You simply send in a photo of yourself, and the company sends back a photo of you skiing, or boating, or night clubbing, or whatever, with a model. Of course, Select-a-Model must pay the supermodels for the use of their images. To anticipate cash flows, the company wants to set up a forecasting system to predict sales. The following table gives monthly demand for the past two years for three of the top-selling models.

Month	Model 1	Model 2	Model 3
1	82	95	148
2	25	12	125
3	44	90	78
4	36	56	53
5	27	54	25
6	91	65	29
7	100	65	9
8	33	92	68
9	97	91	84
10	92	116	110
11	39	141	147
12	94	137	120
13	70	124	147
14	72	90	109
15	90	72	96
16	73	71	70
17	6	92	42
18	30	140	36
19	98	170	34
20	9	150	28
21	0	141	71
22	17	180	102
23	25	171	103
24	11	124	144

a. Plot the demand data for all three models, and suggest a forecasting model that might be suited to each.

b. Find suitable constants for model 1. How good a predictor is the resulting model?

c. Find suitable constants for model 2. How good a predictor is the resulting model?

d. Find suitable constants for model 3. How good a predictor is the resulting model?

5. Can-Do Canoe sells lightweight portable canoes. Quarterly demand for its most popular product family over the past three years has been as follows:

Year	1996				1997				1998			
Quarter	1	2	3	4	1	2	3	4	1	2	3	4
Demand	25	120	40	60	30	140	60	80	35	150	55	90

a. Use an exponential smoothing model with smoothing constant $\alpha = 0.2$ to develop a forecast for these data. How does it fit? What is the resulting MSD?

b. Use an exponential smoothing with a linear trend model with smoothing constants $\alpha = \beta = 0.2$ to develop a forecast for these data. How does it fit? What is the resulting MSD?

c. Use the Winters method with smoothing constants $\alpha = \beta = \gamma = 0.2$ to develop a forecast for these data. How does it fit? What is the resulting MSD?

d. Find smoothing constants that minimize MSD over the second two years of data. How does the resulting forecast fit the data in the third year?

e. Find smoothing constants that minimize MSD over the third year of data. How much better does the model fit the data in the third year than that of part *d*? Which model, *d* or *e*, do you think is likely to better predict demand in year 4?

6. Suppose a plant produces 50 customized high-performance bicycles per day and maintains on average 10 days' worth of WIP in the system.

   a. What is the average cycle time (i.e., time from when an order is released to the plant until the bicycle is completed, ready to ship)?

   b. When would the *conveyor model* predict that the 400th bicycle will be completed?

   c. Suppose we currently have orders for 1,000 bicycles (i.e., including the orders for the 500 bicycles that have already been released to the plant) and a customer is inquiring about when we could deliver an order of 50 bicycles. Use the conveyor model to predict when this new order will be completed. If we have flexibility concerning the due date we quote to the customer, should we quote a date calculated earlier, later, or at the same time as that computed using the conveyor model? Why?

7. Marco, the manager of a contractor's supply store, is concerned about predicting demand for the DeWally 519 hammer drill in order to help plan for purchasing. He has brought in a team of MBAs, who have suggested using a moving-average or exponential smoothing method. However, Marco is not sure this is the right approach because, as he points out, sales of the drill are affected by price. Since the store periodically runs promotions during which the price is reduced, he thinks that price should be accounted for in the forecasting model. The following are price and sales data for the past 20 weeks.

Week	Price	Sales
1	199	25
2	199	27
3	199	24
4	179	35
5	199	21
6	199	26
7	199	29
8	199	28
9	199	32
10	169	48
11	169	45
12	199	30
13	199	38
14	199	37
15	199	38
16	199	39
17	179	45
18	199	40
19	199	39
20	199	42

a. Propose an alternative to a time series model for forecasting demand for the Dewally 519.

b. Use your method for the first n weeks of data to predict sales in week $n + 1$ for $n = 15, \ldots, 19$. How well does it work?

c. What does your model predict sales will be in week 21 if the price is $199? If the price is $179?

8. Suppose Clutch-o-Matic, Inc., has been approached by an automotive company to provide a particular model of clutch on a daily basis. The automotive company needs 1,000 clutches per day, but expects to divide this production among several suppliers. What the company wants from Clutch-o-Matic is a commitment to supply a specific number each day (i.e., a daily quota). Under the terms of the contract, failure to supply the quota will result in a financial penalty.

   Clutch-o-Matic has a line it could dedicate to this customer and has computed that the line has a mean daily production of 250 clutches with a standard deviation of 50 clutches under single (eight-hour) shift production. A clutch sells for $200, of which $30 is profit. If overtime is used, union rules require at least two hours of overtime pay. The cost of worker pay, supervisor pay, utilities, etc., for running a typical overtime shift has been estimated at $6,200.

   a. What is the profit-maximizing quota from the perspective of Clutch-o-Matic?

   b. What is the average daily profit to Clutch-o-Matic if the quota is set at the level computed in *a*?

   c. If the automotive company insists on 250 clutches per day, is it still profitable for Clutch-o-Matic? How much of a decrease in profit does this cause relative to the quota from *b*?

   d. How might a quota-setting model like this one be used in the negotiation process between a supplier and its customers requesting JIT contracts?

CHAPTER

14 SHOP FLOOR CONTROL

Even a journey of one thousand li begins with a single step.
Lao Tzu

14.1 Introduction

Shop floor control (SFC) is where planning meets parts. As such, it is the foundation of a production planning and control system. Because of its proximity to the actual manufacturing process, SFC is also a natural vehicle for collecting data for use in the other planning and control modules. A well-designed SFC module both controls the flow of material through the plant and makes the rest of the production planning system easier to design and manage.[1]

Despite its logical importance in a production planning hierarchy, SFC is frequently given little attention in practice. In part, this is because it is perceived, too narrowly, we think, as purely material flow control. This view makes it appear that once one has a good schedule in hand, the SFC function can be accomplished by routing slips attached to parts and giving the sequence of process centers to be visited; one simply works on parts in the order given by the schedule and then moves them according to the routing slips. As we will see here and in Chapter 15, even with an effective scheduling module, the control of material flow is frequently not so simple. No scheduling system can anticipate random disruptions, but the SFC module must accommodate them anyway. Furthermore, as we have already noted and will discuss further in this chapter, material flow control is simply too narrow a focus for SFC. When one includes the other functions that are appropriately included in SFC, this module assumes a critical function in the overall planning hierarchy.

There may be another reason for the lack of attention to SFC. A set of results from the operations management literature indicates that decisions affecting material flow are less important to plant performance than are decisions dealing with shaping the production environment. Krajewski et al. (1987) used simulation experiments to show

[1] We remind the reader that we are using the term *module* to include all the decision making, record keeping, and computation associated with a particular planning or control problem. So while the SFC module may make use of a computer program, it involves more than this. Indeed, some SFC modules may not even be computerized at all.

that the benefits from improving the production environment by reducing setup times, improving yields, and increasing worker flexibility were far larger than the benefits from switching to a kanban system from a reorder point or MRP system. On the basis of their study, they concluded that (1) reshaping the production environment was key to the Japanese success stories, and (2) if a firm improves the environment enough, it does not make much difference what type of production control system is used. In a somewhat narrower vein, Roderick et al. (1991) used simulation to show that the release rate had a far greater effect on performance than did work sequencing at individual machines. Their conclusion was that master production schedule (MPS) smoothing is likely to have a stronger beneficial effect than sophisticated dispatching techniques for controlling work within the line.

If one narrowly interprets SFC to mean dispatching or flow control between machines, then studies like these do indeed tend to minimize its importance. However, if one takes the broader view that SFC controls flow *and* establishes links between other functions, then the design of the SFC module serves to shape the production environment. For instance, the very decision to install a kanban system evinces a commitment to small-lot manufacture and setup reduction. Moreover, a pull system automatically governs the release rate into the factory, thereby achieving the key benefits identified by Roderick et al.

But is kanban (or something like it) *essential* to achieving these environmental improvements? Krajewski et al. imply that environmental improvements, such as setup reduction, could be just as effective without kanban, while JIT proponents contend that kanban is needed to apply the necessary pressure to force these improvements. Our view is closer to that of the JIT proponents; without an SFC module that promotes environmental improvements and, by means of data collection, documents their effectiveness, it is extremely difficult to identify areas of leverage and make changes stick. Thus, we will take the reshaping of the production environment as part and parcel of SFC module design.

On the basis of our discussions in Chapters 4, 10, and 13, we feel that the most effective (and manageable) production environment is that established by a pull system. Recall that the basic distinction between push and pull is that push systems schedule production, while pull systems authorize production. Fundamental to any pull mechanism for authorizing production is a WIP cap that limits the total inventory in a production line. In our terminology, a system cannot be termed *pull* if it does not establish a WIP cap. Complementing this defining feature are a host of other supporting characteristics of pull systems, including setup time reduction, worker cross-training, cellular layouts, quality at the source, and so on. The manner and extent to which these techniques can be used depend on the specific system. The objective for the SFC module is to make the actual production environment as close as possible to the ideal environments we examined in Chapters 4 and 10. At the same time, the SFC module should be relatively easy to use, integrate well with the other planning functions, and be flexible enough to accommodate changes the plant is likely to face. As we will see, because manufacturing settings differ greatly, the extent to which we can do this will vary widely, as will the nature of the appropriate SFC module.

Figure 14.1 illustrates the range of functions one can incorporate into the SFC module. At the center of these functions is **material flow control (MFC)**, without which SFC would not be shop floor control. Material flow control is the mechanism by which we decide which jobs to release into the factory, which parts to work on at the individual workstations, and what material to move to and between workstations. Although SFC is sometimes narrowly interpreted to consist solely of material flow control, there are a

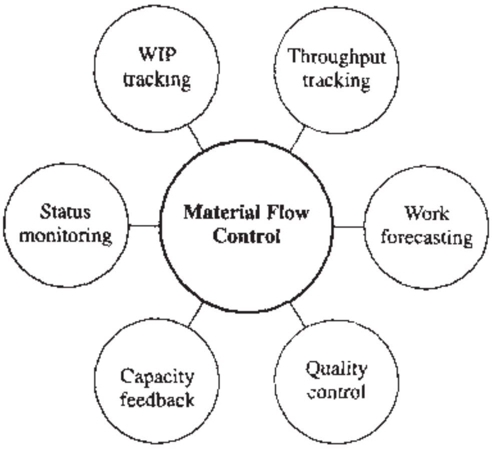


Figure 14.1
Potential functions of the SFC module

number of other functions that are integrally related to material flow control, and a good SFC module can provide platform for these.

WIP tracking, status monitoring, and **throughput tracking** deal with what is happening in the plant in real time. WIP tracking involves identifying the current location of parts in the line. Its implementation can be detailed and automated (e.g., through the use of optical scanners) or rough and manual (e.g., performed by log entries at specified points in the line). Status monitoring refers to surveillance of other parameters in the line besides WIP position, such as machine status (i.e., up or down) or staffing situation. Throughput tracking consists of measuring output from the line or plant against an established production quota and/or customer due dates, and it can be used to anticipate the need for overtime or staffing shifts.

Since the SFC module is the place where real-time control decisions are implemented, it is a natural place for monitoring these types of changes in real-time status of the line. If the SFC module is implemented on a computer, these data collection and display tasks are likely to share files used by the SFC module for material flow control. Even if material flow control is implemented as a manual system, it makes sense to think about monitoring the system in conjunction with controlling it, since this may have an impact on the way paperwork forms are devised. A specific mechanism for monitoring the system is **statistical throughput control (STC),** in which we track progress toward making the periodic production quota. We give details on STC in Section 14.5.1.

In addition to collecting information about real-time status, the SFC module is a useful place to collect and process some information pertaining to the future beyond real time. One possibility is the **real-time simulation** function, in which projections are made about the timing of arrival of specific parts at various points in the line. Chapter 13 addressed this function as an off-line activity. However, it is also possible to incorporate a version of the real-time simulation module directly into the SFC module. The basic mechanism is to use information about current WIP position, collected by the WIP tracking function, plus a model of material flow (e.g., based on the conveyor model) to predict when a particular job will reach a specific workstation. Being able to call up such information from the system can allow line personnel to anticipate and prepare for jobs.

A different function of the SFC module is the collection of data to update capacity estimates. This **capacity feedback** function is important for ensuring that the high-level planning modules are consistent with low-level execution, as we noted in Chapter 13. Since the SFC module governs the movement of materials through the plant, it is the natural place to measure output. By monitoring input over time we can estimate the actual capacity of a line or plant. We will discuss the details of how to do this in Section 14.5.2.

The fact that move points represent natural opportunities for quality assurance establishes a link between the SFC module and **quality control.** If the operator of a downstream workstation has the authority to refuse parts from an upstream workstation on the basis of inadequate quality, then the SFC module must recognize this disruption of a requested transaction. The material flow control function must realize that replacements for rejected parts are required or that rework will cause delays in part arrivals; the WIP tracking function must note that these parts did not move as anticipated; and the work forecasting function must consider the delay in order to make work projections. Furthermore, since quality problems must be noted for these control purposes, it is often convenient to use the system to keep a record of them. These records provide a link to a statistical process control (SPC) system for monitoring quality performance and identifying opportunities for improvement.

In the remainder of this chapter, we give

1. An overview of issues that must be resolved prior to designing an SFC module.
2. A discussion of CONWIP as the basis for an SFC module.
3. Extensions of CONWIP schemes.
4. Mechanisms for tracking production in order to measure progress toward quota in the short term, and collecting and validating capacity data for other planning modules in the long term.

14.2 General Considerations

One is naturally tempted to begin a discussion of the design of an SFC system by addressing questions about the control mechanism itself: Should work releases be controlled by computer? Should kanban cards be used? How do workers know which jobs to work on? And so on. However, even more basic questions should be addressed first. These deal with the general physical and logical environment in which the SFC system must operate.

To develop a reasonable perspective on the management implications of the SFC module, it is important to consider shop floor control from both a *design* and a *control* standpoint. Design issues deal with establishing a system within which to make decisions, while control issues treat the decisions themselves. For instance, choosing a work release mechanism is a design decision, while selecting parameters (e.g., WIP levels) for making the mechanism work is a control issue. We will begin by addressing relatively high-level design topics and will move progressively toward lower-level control topics throughout the chapter.

14.2.1 Gross Capacity Control

Production control systems work best in stable environments. When demand is steady, product mix is constant, and processes are well behaved, almost any type of system (e.g., reorder points, MRP, or kanban) can work well, as shown by the simulation studies of Krajewski et al. (1987). From a manufacturing perspective, we would like to set up production lines and run them at a nice, steady pace without disruptions. Indeed, to a large extent, this is precisely what JIT, with its emphasis on production smoothing and setup reduction, attempts to do. But efforts to create a smooth, easy production environment can conflict with the business objectives to make money, grow and maintain market share, and ensure long-term viability. Customer demand fluctuates, products emerge and decline, technological competition forces us to rely on new and unstable processes. Therefore, while we should look for opportunities to stabilize the environment, we must take care not to lose sight of higher-level objectives in our zeal to do this. We shouldn't forgo an opportunity to gain a strategic edge via a new technology simply because the old technology is more stable and easier to manage.

Even while we respond to market needs, there are things we can do to avoid unnecessary volatility in the plant. One way to stabilize the environment in which the SFC module must operate is to use gross capacity control to ensure that, when running, the lines are close to optimally loaded. The goal is to avoid drastic swings in line speed by controlling the amount of time the line, or part of it, is used. Specific options for gross capacity control include

1. *Varying the number of shifts.* For instance, three shifts per day may be used during periods of heavy demand, but only two shifts during periods of lighter demand. A plant can use this option to match capacity to seasonal fluctuations in demand. However, since it typically involves laying off and rehiring workers, it is only appropriate for accommodating persistent demand changes (e.g., months or more).
2. *Varying the number of days per week.* For instance, weekends can be used to meet surges of demand. Since weekend workers can be paid on overtime, a plant can use this approach on much shorter notice than it can use shift changes. Notice that we are talking here of *planned overtime,* where the weekends are scheduled in advance because of heavy demand. This is in contrast with *emergency overtime* used to make up quota shortfalls, as we discussed in Chapter 13.
3. *Varying the number of hours per day.* Another source of planned overtime is to lengthen the workday, for instance, from 8 to 10 hours.
4. *Varying staffing levels.* In manual operations, capacity can be augmented by adding workers (e.g., floating workers from another part of the plant, or temporary hires). In multimachine workstations, managers can alter capacity by changing the number of machines in use, possibly requiring staffing changes as well.
5. *Using outside vendors.* One way to maintain a steady loading on a plant or line is to divert work beyond a specified level to another firm. Ideally, this transfers at least part of the burden of demand variability to the vendor.[2]

As the term *gross* capacity control implies, these activities can only alter the effective capacity in a rough fashion. Shifts must be added whole and only infrequently removed. Weekend overtime may have to be added in specific amounts (e.g., a day or half-day) due to union rules or personnel policy. Options for varying capacity through floating workers are limited by worker skill levels and loadings in other portions of the plant. Adding and releasing temporary workers requires training and other expenses, which

[2] Of course, there is no guarantee that a vendor will be able to accommodate varying demand any better than the firm itself. Moreover, vendors who can are likely to charge for it. So while vendors can be useful, they are hardly a panacea.

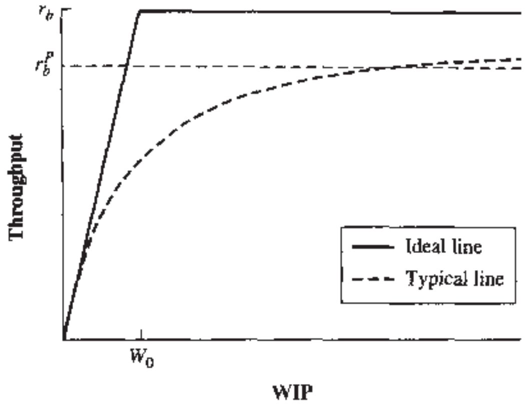


FIGURE 14.2

Throughput as a function of WIP in a single-product line

limits the flexibility of this option. Vendoring contracts may require minimum and/or maximum amounts of work to be sent to the vendor, so this approach may remove only part of the demand variability faced by the firm. Moreover, since finding and certifying vendors is a time-consuming process, vendor contracts are likely to persist over time.

Despite the limitations of the options discussed, it is important that they, or other methods, be used to match capacity to demand at least roughly. Huge variations in the workload of a line will induce tremendous variability throughout the line and will seriously degrade its performance. Kanban or CONWIP requires fairly steady rate-driven lines. We will discuss a pull alternative for lines that cannot achieve this type of stability via gross capacity control. However, no system can entirely mitigate the negative effects of highly variable demand.

14.2.2 Bottleneck Planning

In Part II we stressed that the rate of a line is ultimately determined by the bottleneck, or slowest, process. In the simple single-product, single-routing lines we considered in Chapter 7 to illustrate basic factory dynamics, the bottleneck process represents the maximum rate of the line. This rate is only achieved when the WIP in the line is allowed to become large,[3] as illustrated in Figure 14.2.

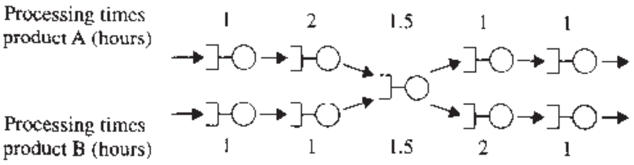
In lines where all parts follow the same routing and processing times are such that the same process is the slowest operation for all parts, the conveyor model is an accurate representation of reality and useful for analysis, as well as intuition. In such cases, the bottleneck plays a key role in the performance of the line and therefore should be given special attention by the SFC module. Because throughput is a direct function of the utilization of the bottleneck, it makes sense to trigger releases into the line according to the status of the bottleneck. Such "pull from the bottleneck" schemes can work well in some systems, and we will discuss them further.

In spite of the theoretical importance of bottlenecks, it has been our experience that few manufacturers can identify their bottleneck process with any degree of confidence. The reason is that few manufacturing environments closely resemble a single-product, single-routing line. Most systems involve multiple products with different processing times. As a result, the bottleneck machine for one product may not be the bottleneck for

[3] What is meant by *large*, of course, depends on the amount of variability in the line, as we noted in Chapter 9.

FIGURE 14.3

Routings with a shared resource


Processing times product A (hours)
1
2
1.5
1
1
Processing times product B (hours)
1
1
1.5
2
1

another product. This can cause the bottleneck to "float," depending on the product mix. Recall that Figure 10.9 illustrated this type of behavior with an example where machine 2 is the bottleneck for product A, machine 4 is the bottleneck for product B, and machine 3 is the bottleneck for a 50-50 mix of A and B.

Multiproduct systems also often involve different routings for different products. For instance, Figure 14.3 shows a two-routing system with a single shared workstation. Whether or not machine 3 is the bottleneck for product A depends on the volume of product B. Similarly, the bottleneck for product B depends on the volume of product A. Thus, the bottlenecks in this system can also float depending on product mix. Furthermore, if the two product lines are under separate management, the location of the bottleneck in each line may be outside the control of the line manager.

This discussion has two important implications for design of the SFC module:

1. *Stable bottlenecks are easier to manage.* A line with a distinct identifiable bottleneck is simpler to model (i.e., with the conveyor model) and control than a line with multiple moving bottlenecks. A manager can focus on the status of the bottleneck and think about the rest of the line almost exclusively in terms of its impact on the bottleneck (i.e., preventing starvation or blocking of the bottleneck). If we are fortunate enough to have a line with a distinct bottleneck, we should exploit this advantage with an SFC module that gives the bottleneck favorable treatment and provides accurate monitoring of its status.

2. *Bottlenecks can be designed.* Although some manufacturing systems have their bottleneck situation more or less determined by other considerations (e.g., the capacity of all key processes would be too expensive to change), we can often proactively influence the bottleneck. For instance, we can reduce the number of potential bottlenecks by adding capacity at some stations to ensure that they virtually never constrain throughput. This may make sense for stations where capacity is inexpensive.[4] Or interacting lines can be separated into cells; for example, the two lines in Figure 14.3 could be separated by adding an additional machine 3 (or dedicating machines to lines, if station 3 is a multimachine workstation). This type of cellular manufacturing has become increasingly popular in industry, in large part because small, simple cells are easier to manage than large, complex plants.

Although it is difficult to estimate accurately the cost benefits of simplifying bottleneck behavior, it is clear that there *are* costs associated with complexity. The simplest plant to manage is one with separate routings and distinct, steady bottlenecks. Any departures from this only serve to increase variability, congestion, and inefficiency. This does not mean that we should automatically add capacity until our plant resembles this ideal; only that we should consider the motivation for departures from it. If we are

[4] Note that the idea of deliberately adding capacity that will result in some resources being underutilized runs counter to the principle of line balancing. Economic justification of unbalancing the line requires taking a linewide perspective that considers variability, as we have stressed throughout this book.

plagued by a floating bottleneck that could be eliminated via inexpensive capacity, the addition deserves consideration. If interacting routings could be separated without large cost, we should look into it.

Moreover, line design and capacity allocation need not be plantwide to be effective. Sometimes great improvements can be achieved by assigning a few high-volume product families to separate, well-designed cells, leaving many low-volume families to an inefficient job shop portion of the plant. This "factory within a factory" idea has been promoted by various researchers and practitioners, most prominently Wickham Skinner (1974), as part of the **focused factory** philosophy. The main idea behind focused factories is that plants can do only a few things very well and therefore should be focused on a narrow range of products, processes, volumes, and markets. As we will see repeatedly throughout Part III, simplicity offers substantial benefits throughout the planning hierarchy, from low-level shop floor control to long-range strategic planning.

14.2.3 Span of Control

In Chapter 13, we discussed disaggregation of the production planning problem into smaller, more manageable units. We devoted most of that discussion to disaggregation along the time dimension, into short-, intermediate-, and long-range planning. But other dimensions can be important as well. In particular, in large plants it is essential to divide the plant by product or process in order to avoid overloading individual line managers.

Typically, a reasonable **span of control,** which usually refers to the number of employees under direct supervision of the manager, is on the order of 10 employees. A line with many more workers than this will probably require intermediate levels of management (foremen, lead technicians, multiple layers of line managers). Of course, 10 is only a rough rule of thumb; the appropriate number of employees under direct supervision of a manager will vary across plants. Strictly speaking, the term *span of control* should really refer to more than simply the number of subordinates, to consider the range of products or processes the manager must supervise.

For instance, printed-circuit board (PCB) manufacture involves, among other operations, a lamination process, in which copper and fiberglass sheets are pressed together, and a circuitize process, in which the copper sheets are etched to produce the desired circuitry. The technology, equipment, and logistics of the two processes are very different. Lamination is a batch process involving large mechanical presses, while circuitizing is a combination of a one-board-at-a-time process using optical expose machines and a conveyorized flow process involving chemical etching. These differences, along with physical separation, make it logical to assign different managers to the two processes.

How a line is broken up, for bottleneck design, span of control, or other considerations, is relevant to the configuration of the SFC module. Depending on the complexity of the line, managers may be able to coordinate movement of material through the portion of the line for which they are responsible, with very little assistance from the production control system. But the managers cannot coordinate activities outside their areas. Presumably, a higher-level manager has responsibility spanning disparate portions of the plant (and, of course, the plant manager has ultimate responsibility for the whole plant). However, at a real-time control level, these higher-level managers cannot force coordination. This must be done by the line managers, using information provided by the SFC module.

At a minimum, the SFC module must tell managers what parts are required by downstream workstations. If the module can also project what materials will be arriving at each station, so much the better, since this information enables the line managers to plan their activities in advance. The division of the line for management purposes provides a natural set of points in the line for reporting this information. How the line is divided may also affect the other functions of the SFC module listed in Figure 14.1. For purposes of accountability, it may be desirable to build in quality checks between workstations under separate management (e.g., the downstream station checks parts from an upstream station and refuses to accept them if they do not meet specifications). Under these conditions, the links between SFC and quality control must be made with this in mind.

14.3 CONWIP Configurations

As we observed in Chapter 5, JIT authors sometimes get carried away with the rhetoric of simplicity, making statements like "Kanban ... can be installed ... in 15 minutes, using a few containers and masking tape" (Schonberger 1990, p. 308). As any manager who has installed a pull system knows, getting a system that works well is *not* simple or easy. Manufacturing enterprises are complex, varied activities. Neither the high-level philosophical guidelines of "romantic JIT" nor the collection of techniques from "pragmatic JIT" can possibly provide ready-made solutions for individual manufacturing environments. With this in mind, we begin our review of possible SFC configurations. We start with the simplest possibilities, note where they will and won't work well, and move to more sophisticated methods for more complex environments. Since we cannot discuss every option in detail, our hope is that the range offered here will provide the reader with a mix-and-match starting point for choosing and developing SFC modules for specific applications.

14.3.1 Basic CONWIP

The simplest manufacturing environment from a management standpoint is the single-routing, single-family production line. If the following conditions hold, then this model is an accurate approximation of reality, and a basic CONWIP system (where releases are coordinated with completions to hold the WIP level in the line constant) will work well in the SFC module, for the reasons discussed in Chapter 10:

1. There are *constant routings* so that all parts traverse the same sequence of machines. Actually, if some parts contain a few extra operations (e.g., installation of a deluxe feature) that do not substantially alter flow time, we may be able to ignore this and still use basic CONWIP. However, if routings are conditional (e.g., jobs may be diverted to a rework line or sent out to a vendor), then we may not be able to treat the line as a single routing and will require more than basic CONWIP.
2. *Processing times are similar* so that all parts require roughly the same amount of time at each process center. This implies that the bottleneck will be stable. We do not require that the bottleneck be sharply defined (i.e., significantly slower than other machines), however.
3. There are *no significant setups* so that the time through the line for an individual job is not strongly affected by the sequence of jobs.
4. There are *no assemblies*, so we can view the progression of jobs as a linear flow. We will modify basic CONWIP to accommodate assemblies later.

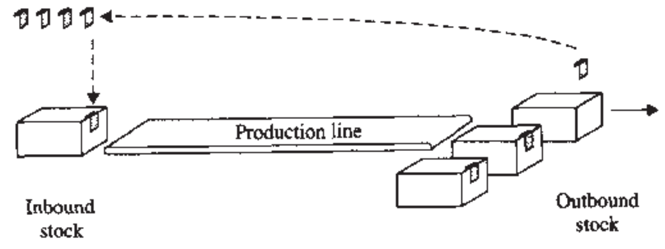
Perhaps the simplest way to maintain the constant-WIP protocol is by means of physical cards or containers, as Figure 14.4 illustrates. Raw materials arrive to the line in standard containers but are only released into the line if there is an available CONWIP card. These cards can be laminated sheets of paper, metal or plastic tags, or the empty containers themselves. Since no routing or product information is required on the cards, they can be very simple. Provided that work is only released into the line with a card, and cards are faithfully recycled (they don't get trapped with a job diverted for rework or terminated by an engineering change order), the WIP in the line will remain constant at the level set by the number of CONWIP cards.

Even in this simple system there are SFC issues to resolve.

1. *Work backlog.* Because the CONWIP cards do not contain product information, a line manager or operator needs additional information to select jobs to release into the line. This is the task of the sequencing and scheduling module, which may use a simple earliest due date (EDD) sequence (because of the no-setup assumption) or a more involved batching routine (to achieve a rhythm by working on similar parts for extended periods). Once generated, the backlog can be communicated to the line in a variety of ways. The simplest consists of a piece of paper with a prioritized list of jobs. Whenever a CONWIP card is available, the next job for which raw materials are available is released into the line. Some situations may call for more sophisticated work backlog displays, for example, showing priorities or projected arrival times.

2. *Line discipline.* In general, a line should maintain a first-in-system first-out order. This means that, barring yield loss, rework problems, or passing at multimachine stations, the jobs will exit the line in the same order they were released. Since the CONWIP protocol keeps the line running at a steady pace, this makes it easy to predict when jobs—even those still on the work backlog—will be completed. However, if the CONWIP line is long, there may arise situations in which certain jobs require expediting. While we wish to discourage incautious use of expediting because it can dramatically increase variability in the line, it is unreasonable to expect the firm never to expedite. To minimize the resulting disruption, it may make sense to allow only two levels of priority and to establish specific **passing points.** The passing points are buffers or stock points in the line, typically between segments run as CONWIP loops, where "hot" jobs are allowed to pass "normal" jobs. The discipline of a workstation taking material from such a buffer is to take the first job from the **hot list,** if there is one, and, if not, the oldest job currently in the buffer. To allow passing only at designated points in the line makes it easier to build a model (the real-time simulation module) for predicting when jobs will exit the line. If many levels of priority and unrestricted passing are permitted, the variability or "churn" in the line can become acute, and it can be almost impossible to predict line behavior.

Figure 14.4
A CONWIP line using cards



3. *Card counts.* To be effective, a CONWIP SFC module must fix a reasonable WIP level. As we noted in Chapter 13, setting card counts is a function that should be done infrequently (e.g., monthly or quarterly), not in real time with the work releases. If CONWIP is being implemented on an established line, the easiest approach for setting card counts is to begin with a count that fixes WIP at the historical level. After the line has stabilized, look at the workstations for persistent queues. If a station's queue virtually never empties, then reducing the card count will not have much effect on throughput and therefore should be done. Make periodic reviews of queue lengths to adjust card counts to accommodate physical changes (hopefully improvements) in the line. If CONWIP is being implemented on a new line, then a reasonable approach is to select the WIP level by choosing a reasonable and *feasible* cycle time CT and estimating the practical production rate of the line r_b^P. Then, using Little's law, set the WIP level as:

$$\text{WIP} = \text{CT} \times r_b^P$$

Assuming actual throughput is close to r_b^P, this will set WIP levels appropriately, provided that CT is actually feasible. Care must be taken not to get overly optimistic about cycle time, since it will lead to an underestimate of the required WIP and therefore a reduction in throughput.

4. *Card deficits.* If the card count is sufficiently large relative to variability in the line, rigidly adhering to the CONWIP protocol can work well. However, there are situations where we may be tempted to violate the constant-WIP release rule. Figure 14.5 illustrates one such situation, where a nonbottleneck machine downstream from the bottleneck is experiencing an unusually long failure, causing the bottleneck to starve for lack of cards. If the nonbottleneck machine is substantially faster than the bottleneck, then it will easily catch up once it is repaired. But in the meantime, we are losing valuable time at the bottleneck. One remedy for this situation is to run a **card deficit,** in which we release some jobs without CONWIP cards into the line. This will allow the bottleneck to resume work. Once the failure situation is resolved, we revert to the CONWIP rules and only allow releases with cards. The jobs without cards will eventually clear the line, and WIP will fall back to the target level. Another remedy for this type of problem is to pull from the bottleneck instead of the end of the line. We discuss this in Section 14.4.2.

5. *Work ahead.* One of the benefits of CONWIP that we identified in Chapter 10 is its ability to opportunistically work ahead of schedule when events permit. For instance, if the bottleneck is unusually fast or reliable this week, we may be able to do more work than we had planned. Assuming that the master production schedule is full, it probably makes sense to take advantage of our good fortune—up to a limit. While it almost certainly makes sense to start some of next week's jobs, it may not make sense to start jobs that are not due for months. If the MPS for a particular routing is not full, which is a real possibility in a plant with many routings, each of which is used sporadically, then we may want to establish a **work-ahead window.**

For instance, when authorized by the CONWIP mechanism, we may release the next job into the line, *provided that it is within n weeks of its due date*. Setting the limit *n* is an

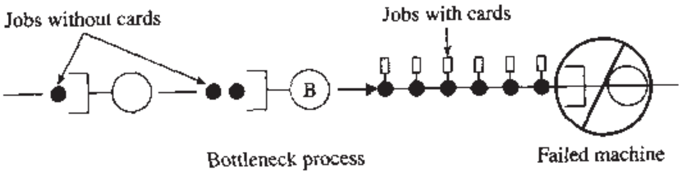


Figure 14.5
CONWIP card deficits in failure situations

additional CONWIP design question, which is closely related to the concepts of frozen zones and time fences discussed in Chapter 3. Since jobs within the frozen zone of their due dates are not subject to change, it makes sense to allow CONWIP to work ahead on them. Jobs beyond the restricted frozen zone (or partially restricted time fences) are much riskier to work ahead on, since customer requirements for these jobs may change. Clearly, the choice of an appropriate work-ahead policy is strongly dependent on the manufacturing environment.

14.3.2 Tandem CONWIP Lines

Even if we satisfy the conditions for basic CONWIP to be applicable (constant routings, similar processing times, no significant setups, and no assemblies), we may not want to run the line as a single CONWIP loop. The reason is that span-of-control considerations may encourage us to decouple the line into more manageable parts. One way to do this is to control the line as several tandem CONWIP loops separated by WIP buffers. The WIP levels in the various loops are held constant at specified levels. The interloop buffers hold enough WIP to allow the loops to temporarily run at different speeds without affecting (blocking or starving) one another. This makes it easier for different managers to be in charge of the different loops. The extra WIP and cycle time introduced by the buffers also degrade efficiency. This is a tradeoff one must evaluate in light of the particular needs of the manufacturing system.

Figure 14.6 illustrates different CONWIP breakdowns of a single production line, ranging from treating the entire line as a single CONWIP loop to treating each workstation as a CONWIP loop. Notice that this last case, with each workstation as a loop, is identical to one-card kanban. In a sense, basic CONWIP and kanban are extremes in a continuum of CONWIP-based SFC configurations. The more CONWIP loops we break the line into, the closer its behavior will be to kanban. As we discussed in Chapter 10, kanban provides tighter control over the material flow through individual workstations and, if WIP levels are low enough, can promote communication between adjacent stations. However, because there are more WIP levels to set in kanban, it tends to be more complex to implement than basic CONWIP. Therefore, in addition to the efficiency/span-of-control tradeoff to consider in determining how many CONWIP loops to use to control a line, we should think about the complexity/communication tradeoff.

Another control issue that arises in a line controlled with multiple tandem CONWIP loops concerns when to release cards. The two options are (1) when jobs enter the interloop buffers or (2) when they leave them. If CONWIP cards remain attached to jobs in the buffer at the end of a loop, then the sum of the WIP in the line plus the WIP in the buffer will remain constant. Therefore, if WIP in the buffer reaches the level specified by the card count, then the loop will shut down until the downstream loop removes WIP from the buffer and releases some cards. As Figure 14.7 illustrates (in loops 1 and 3), this mechanism makes sense for nonbottleneck loops that are fast enough to keep pace with the overall line. If we did not link loop 1 to the pace of the line by leaving cards attached to jobs in the buffer, it could run far ahead of other loops, swamping the system with WIP.

If one loop is a clearly defined bottleneck, however, we may want to decouple it from the rest of the line, in order to let it run as fast as it can (i.e., to work ahead). As loop 2 in Figure 14.7 illustrates, we accomplish this by releasing cards as soon as jobs exit the end of the line—before they enter the downstream buffer. This will let the loop run as fast as it can, subject to availability of WIP in the upstream buffer and subject to a WIP cap on the total amount of inventory that can be in the line at any point in time.

FIGURE 14.6
Tandem CONWIP loops

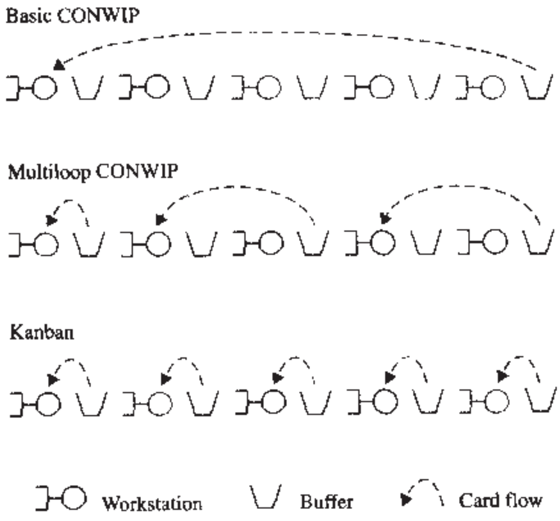
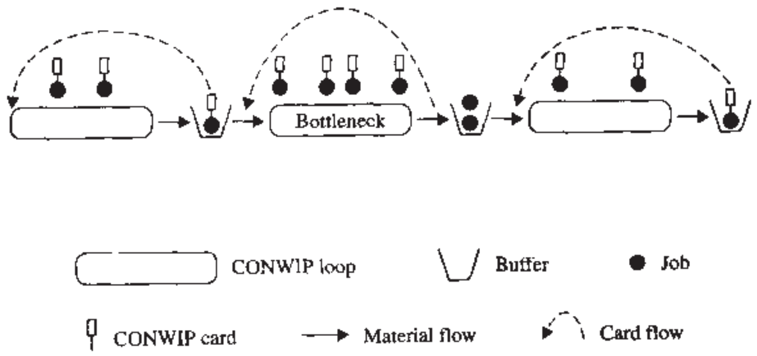


FIGURE 14.7
Coupled and uncoupled CONWIP loops



Of course, this means that the WIP in the downstream buffer can float without bound, but as long as the rest of the line is consistently faster than the bottleneck loop, the faster portion will catch up and therefore WIP will not grow too large. Of course, in the long run, all the CONWIP loops will run at the same speed, that set by the bottleneck loop.

14.3.3 Shared Resources

While it is certainly simplest from a logistics standpoint if machines are dedicated to routings—and this is precisely what is sometimes achieved by assigning a set of product families to manufacturing cells—other considerations sometimes make this impossible. For instance, if a certain very expensive machine is required by two different products with otherwise separate routings, it may not be economical to duplicate the machine in order to completely separate the routings. The result will be something like that illustrated previously in Figure 14.3. If several multiple resources are shared across many routings, the situation can become quite complex.

Shared resources complicate both control and prediction of CONWIP lines. Control is complicated at a shared resource because we must choose a job to work on from multiple incoming routings. If the shared resource is in the interior of a CONWIP loop, then the natural information to use for making this choice is the "age" of the incoming jobs. The proper choice is to work on jobs in FISFO (first-in-system first-out) order, because the time a job entered the line corresponds to the time of a downstream demand, as it is a pull system. Hence FISFO will coordinate production with demand.

If it is important to ensure that the shared resource works on jobs imminently needed downstream, then it may make sense to break the line into separate CONWIP loops before and after the shared resource, as Figure 14.8 illustrates. This figure shows two routings, for product families A and B, that share a common resource. Both routings are treated as CONWIP loops before and after the common resource. This provides the common resource with incoming parts in the upstream buffers, and with cards indicating downstream replenishment needs. Working on jobs whose cards have been waiting longest (provided there are appropriate materials in the incoming buffer) is a simple way to force the shared resource to work on parts most likely to be needed soon. If a machine setup is required to switch between families, then an additional rule about how many parts of one family to run before switching may be required.

Shared resources also complicate prediction. While the conveyor model can be quite accurate for estimating the exit times of jobs from a single CONWIP line, it is not nearly as accurate for a line with resources shared by other lines. The reason is that the outputs from one line can strongly depend on what is in the other lines. A simple way to adapt the conveyor model to approximate this situation is to preallocate capacity. For example, suppose two CONWIP lines, for product families A and B, share a common resource, where on average family A utilizes 60 percent of the time of this resource and family B utilizes 40 percent. Then we can roughly treat the line for family A by inflating the process times on the shared resource by dividing them by 0.6 to account for the fact that the resource devotes only 60 percent of its time to family A. Likewise, we treat the line for family B by dividing processing times on the shared resource by 0.4.

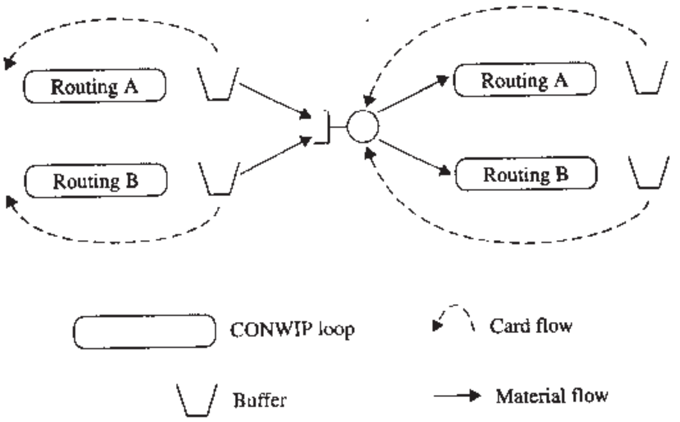
To illustrate this analysis in a little greater detail, suppose that the shared resource in Figure 14.8 requires one hour per job on routing A and two hours per job on routing B. If 60 percent of the jobs processed by this resource are from routing A and 40 percent are from B, then the fraction of processing hours (hours spent running product) that are devoted to A is given by

$$\frac{1 \times 0.6}{1 \times 0.6 + 2 \times 0.4} = 0.4286$$

Therefore, the fraction of processing hours devoted to B is $1 - 0.4286 = 0.5714$. The 42.86 percent number is very much like an *availability* caused by machine outages. In

FIGURE 14.8

Splitting a CONWIP loop at a shared resource



effect, the resource is available to A only 42.86 percent of the time. Thus, while the rate of the shared resource would be one job per hour if only A parts were run, it is reduced to 1×0.4286 job per hour due to the sharing with B. The average processing time is the inverse of this rate, or $1/0.4286 = 2.33$ hours per job. Similarly, the average processing of a B job is

$$\frac{2}{0.5714} = 3.50 \text{ hours per job}$$

Using these inflated processing times for the shared resource, we can now treat routings A and B as entirely separate CONWIP lines for the purposes of analysis. Of course, if the volumes on the two routings fluctuate greatly, then the output times will vary substantially above and below those predicted by the conveyor model. The effect will be very much the same as having highly variable (e.g., long infrequent, as opposed to short frequent) outage times on a resource in a CONWIP line. Therefore, if we use such a model to quote due dates, we have to add a larger inflation factor to compensate for this extra variability.

14.3.4 Multiple-Product Families

We now begin relaxing the assumptions needed to justify basic CONWIP by considering the situation where the line has multiple-product families. We still assume a simple flow line with constant routings and no assemblies, but now we allow different product families to have substantially different processing times and possibly sequence-dependent setups. Under these conditions, it may no longer be reasonable to fix the WIP level in a CONWIP loop by holding the number of units in the line constant. The reason is that the total workload in the line may vary greatly due to the difference in processing times across products. It may make more sense to adjust the WIP count for capacity.

One plausible measure of the work in the system would be hours of processing time at the bottleneck machine. Under this approach, if a unit of product A requires one hour on the bottleneck and B requires two hours, then when one unit of B departs the line, we allow two units of product A to enter (provided that it is next on the work backlog). As long as the location of the bottleneck is relatively insensitive to product mix, this mechanism will tend to maintain a stable workload at the bottleneck. If the bottleneck changes with mix (i.e., different products have different machines as their slowest resource), then computing a capacity-adjusted WIP level is more difficult. We could use total hours of processing time on all machines. However, we will probably need a higher WIP level than would be required for a system with a stable bottleneck, to compensate for the variability caused by the moving bottleneck. Furthermore, if the total processing times of different products do not vary much, this approach will not be much different from the simpler approach of counting WIP in physical units.

If we count WIP in capacity-adjusted standard units, it becomes more difficult to control the WIP level with a simple mechanism like cards. Instead of trying to attach multiple cards to jobs to reflect their differing complexity, it probably makes sense to use an electronic system for monitoring WIP level. Figure 14.9 illustrates an electronic **CONWIP controller,** which consists of a local-area network (LAN) with computers located at the front and back of the line. The computers monitor the adjusted WIP level and indicate when it falls below the target level (e.g., by changing an indicator light from red to green). When this happens, the operator of the first workstation selects the next job on the work backlog for which the necessary materials are available (displayed on the computer terminal as showing the due date, DD, part number, PN, and quantity to be

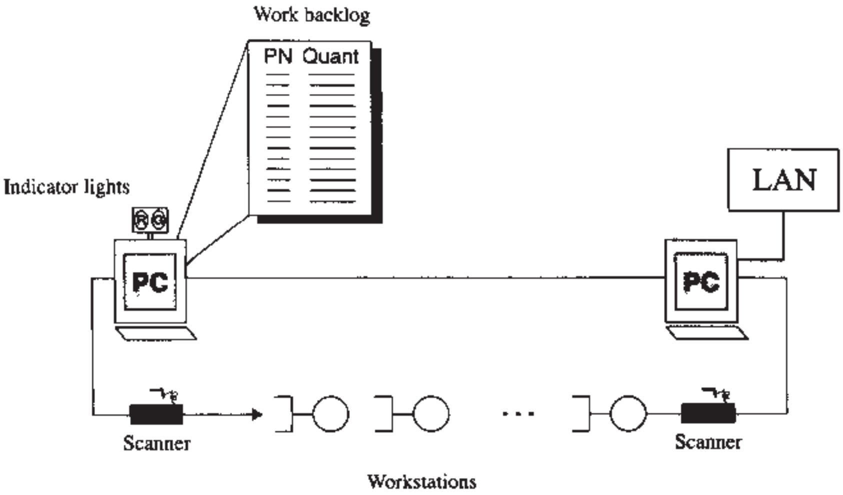


Figure 14.9
A CONWIP line using electronic signals

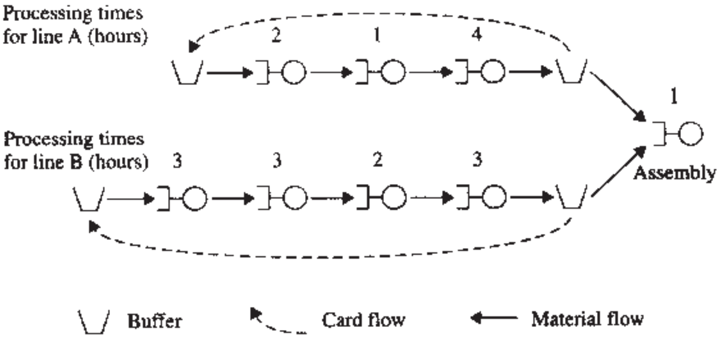

Processing times for line A (hours)
2
1
4
1
Processing times for line B (hours)
3
3
2
3
Assembly
Buffer
Card flow
Material flow

Figure 14.10
CONWIP control of an assembly process

released, (Quant)) and releases it into the line. This release is recorded by keyboard or optical scanner and is added to the capacity-adjusted WIP level. At the end of the line, job outputs are also recorded and subtracted from the WIP level. Exceptions, such as fallout due to yield loss, may also need to be recorded on one of the computer terminals.

14.3.5 CONWIP Assembly Lines

We now further extend the CONWIP concept to systems with assembly operations. Figure 14.10 illustrates the simple situation in which an assembly operation is fed by two fabrication lines. Each assembly requires one subcomponent from family A and one subcomponent from family B. The assembly operation cannot begin until both subcomponents are available. The two fabrication lines are controlled as CONWIP loops with fixed, but not necessarily identical, WIP levels. Each time an assembly operation is completed, a signal (e.g., CONWIP card or electronic signal) triggers a new release in each fabrication line. As long as a FISFO protocol is maintained in the fabrication lines, the final assembly sequence will be the same as the release sequence.

Notice that assembly completions need not trigger releases of subcomponents destined for the same assembly. If line A has a WIP level of 9 jobs and line B has a WIP level of 18 jobs, then the release authorized by the next completion into line A will bc used 9 assemblies from now, while the release into line B will be used 18 assemblies from now. If the total process time for line B is longer than that for line A, this type of imbalance makes sense. In general, the longer line will require a larger WIP level (i.e., because of Little's law). Determining precise WIP levels is a bit trickier. Fortunately, performance is robust in WIP level, provided that the lines have sufficient WIP to prevent excessive starvation of the bottleneck.

To illustrate a mechanism for setting ballpark WIP levels in an assembly system, consider the data given in Figure 14.10. Notice that the systemwide bottleneck is machine 3 of line A. Hence, the bottleneck rate is $r_b = 0.25$ job per hour. If we look at the two lines, including assembly, as separate fabrication lines, we can use the critical WIP formula from Chapter 7 on each line. This shows that the WIP levels under ideal (i.e., perfectly deterministic) conditions need to be

$$W_0^A = r_b T_0^A = \tfrac{1}{4}(2+1+4+1) = \tfrac{8}{4} = 2$$
$$W_0^B = r_b T_0^B = \tfrac{1}{4}(3+3+2+3+1) = \tfrac{12}{4} = 3$$

to achieve full throughput. Of course, in reality, there will be variability in the line, so the WIP levels will need to be larger than this. How much larger depends on how much variability there is in the line.

For a line corresponding to the practical worst case discussed in Chapter 7, we can compute the WIP level required to achieve throughput equal to 90 percent of the bottleneck rate by setting the throughput expression equal to $0.9r_b$ and solving for the WIP level w:

$$\frac{w}{W_0 + w - 1} r_b = 0.9 r_b$$
$$\frac{w}{W_0 + w - 1} = 0.9$$
$$w = 0.9(W_0 + w - 1)$$
$$w = 9W_0 - 9 = 9(W_0 - 1)$$

Inflating W_0^A and W_0^B according to this formula yields

$$w^A = 9(2-1) = 9$$
$$w^B = 9(3-1) = 18$$

Unless the line is highly variable, these WIP levels are probably reasonable starting points, from which a process of adjustment can be initiated. If the processing times on all the machines are less variable than the practical worst case (i.e., they have coefficients of variation smaller than one), then the line may operate effectively with smaller WIP levels than this. If the processing times on some machines are more variable than the practical worst case (i.e., they have coefficients of variation larger than one, due to long failures or setups, for example), then even more WIP than this may be required to achieve a reasonable throughput rate.

14.4 Other Pull Mechanisms

We look upon CONWIP as the first option to be considered as an SFC platform. It is simple, predictable, and robust. Therefore, unless the manufacturing environment is such that it is inapplicable, or another approach is likely to produce substantially better performance, CONWIP is a good, safe choice. By using the flexibility we discussed above to split physical lines into multiple CONWIP loops, one can tailor CONWIP to the needs of a wide variety of environments. But there are situations in which a suitable SFC module, while still a pull system, is not what we would term CONWIP. We discuss some possibilities below.

14.4.1 Kanban

As we noted earlier, kanban can be viewed as tandem CONWIP loops carried to the extreme of having only a single machine in each loop. So from a CONWIP enthusiast's perspective, kanban is just a special case of CONWIP. However, Ohno's book contains a diagram of a kanban system that looks very much like a set of CONWIP loops feeding an assembly line. Therefore, the developers of kanban may well have considered CONWIP a form of kanban. As far as we are concerned, this distinction is a matter of semantics; kanban and CONWIP are obviously closely related. The important question concerns when to use kanban (single-station loops) instead of CONWIP (multistation loops).

Kanban offers two potential advantages over CONWIP:

1. By causing each station to pull from the upstream station, kanban may force better interstation communication. Although there may be other ways to promote the same communication, kanban makes it almost automatic.
2. By breaking the line at every station, kanban naturally provides a mechanism like that illustrated in Figure 14.8, for sharing a resource among different routings.

However, kanban also has the following potential disadvantages:

1. It is more complex than CONWIP, requiring specification of more WIP levels. (However, recall that pull systems are fairly insensitive to WIP level. Hence, the WIP levels in kanban need not be set precisely for the system to function well, and therefore this increase in complexity may not be a major obstacle to kanban in most settings.)
2. It induces a tighter pacing of the line, giving operators less flexibility for working ahead and placing considerable pressure on them to replenish buffers quickly.
3. The use of product-specific cards means that at least one standard container of each part number must be maintained at each station, to allow the downstream stations to pull what they need. This makes it impractical for systems with numerous part numbers.
4. It cannot accommodate a changing product mix (unless a great deal of WIP is loaded into the system) because the product-specific card counts rigidly govern the mix of WIP in the system.
5. It is impractical for small, infrequent orders (onesies and twosies). Either WIP would have to be left unused on the floor for long spans of time (i.e., between orders), or the system would be unresponsive to such orders because authorizations signaled by the kanban cards would have to propagate all the way to the beginning of the line to trigger new releases of WIP.

There is little one can do to alleviate the first two disadvantages; complexity and pressure are the price one pays for the additional local control of kanban. However, the remaining disadvantages are a function of product-specific cards and therefore can be mitigated by using **routing-specific cards** and a **work backlog.** Figure 14.11 shows a kanban system with different-color cards for different routings. When a standard

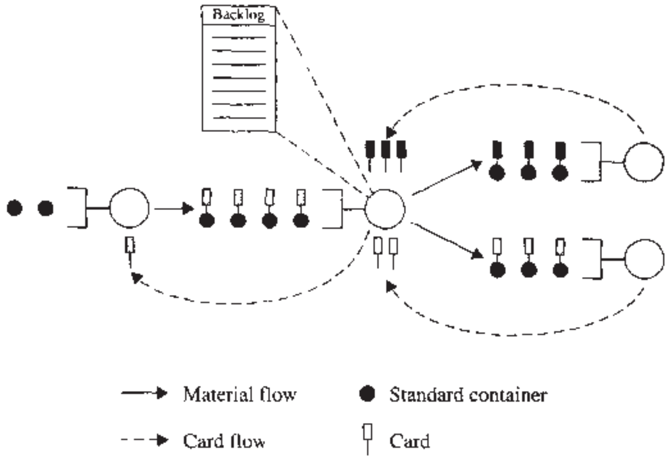


FIGURE 14.11
Kanban with route-specific cards and a work backlog

container is removed from the outbound stock point, the card authorizes production to replace it. The identity of the part that will be produced is determined by the work backlog, which must be established by the sequencing and scheduling module. If a part does not appear on the backlog for an extended period, then it will not be present in the line. The modification of route-specific (as opposed to part-specific) cards enables this approach to kanban to be used in systems with many part numbers.

On the basis of this discussion, it would appear that kanban is best suited to systems with many routings that share resources, especially if products and routings are frequently added and removed. If we are going to break the line into many CONWIP loops to make control of the shared resources easier, then moving all the way to kanban will not significantly change performance. Moreover, if a new routing converts a previously unshared resource to a shared resource, then a kanban configuration will already provide the desired break in the line.

On the other hand, if the various routings have few shared resources and new products tend to follow established routings, there would seem to be little incentive to incur the additional complexity of kanban. The system will probably function more simply and effectively under CONWIP, possibly broken into separate loops for span-of-control reasons, to give special treatment to a shared resource, or to feed buffers at assembly points.

14.4.2 Pull-from-the-Bottleneck Methods

Two problems that can arise with CONWIP (or kanban) in certain environments are the following:

1. *Bottleneck starvation* due to downstream machine failures. As we illustrated in Figure 14.5, we may want to allow releases beyond those authorized by cards to compensate for this situation.
2. *Premature releases* due to the requirement that the WIP level be held constant. Even if a part will not be needed for months, a CONWIP system may trigger its release because WIP in the loop has fallen below its target level. This can reduce flexibility for no good reason (e.g., engineering changes or changes in

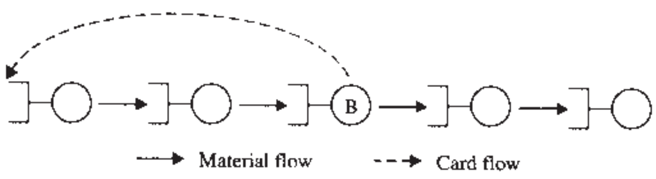


FIGURE 14.12
A pull-from-bottleneck system

customer needs are much more difficult to accommodate once a job has been released to the floor).

We can modify CONWIP to address these situations. The basic idea is to devise a mechanism for enabling the bottleneck to work ahead, but at the same time provide a means of preventing it from working too far ahead. The techniques we will introduce are related to the technique termed **drum-buffer-rope (DBR)** developed by Goldratt (Goldratt and Fox 1986), although he presented DBR primarily as a scheduling methodology rather than an SFC mechanism.

We begin with the simplest version of the **pull-from-bottleneck (PFB)** strategy. Figure 14.12 shows such a system for a single line. This mechanism differs from CONWIP in that the WIP level is held constant in the machines up to and including the bottleneck, but is allowed to float freely past the bottleneck. Since machines downstream from the bottleneck are faster on average than the bottleneck, WIP will not usually build up in this portion of the line. However, if a failure in one of these machines causes a temporary buildup of WIP, it will not cause the bottleneck to shut down, as can occur under CONWIP if card deficits are not used. Therefore, a PFB approach may make sense as an alternative to card deficits in a line with a stable bottleneck. If the bottleneck shifts depending on product mix, then it is not clear where the pulling point should be located, and therefore one may be just as well off pulling from the end of the line (i.e., using regular CONWIP), possibly with a card deficit policy.

The simple PFB approach of Figure 14.12 can mitigate the bottleneck starvation problem associated with CONWIP, but does not address the issue of premature releases. When we are talking about a single line, we often speak as though the line will be kept running at close to full capacity. And this is frequently true in plants with few routings. But in plants with many routings (e.g., a plant tending toward a job shop configuration), some routings may not be used for substantial periods of time. For instance, we have seen plants with 5,000 distinct routings, only a relative few of which contained WIP at any given time. Clearly, under these conditions we do not want to maintain a constant WIP level along the routing, since this would result in releasing jobs that are not needed until far in the future.

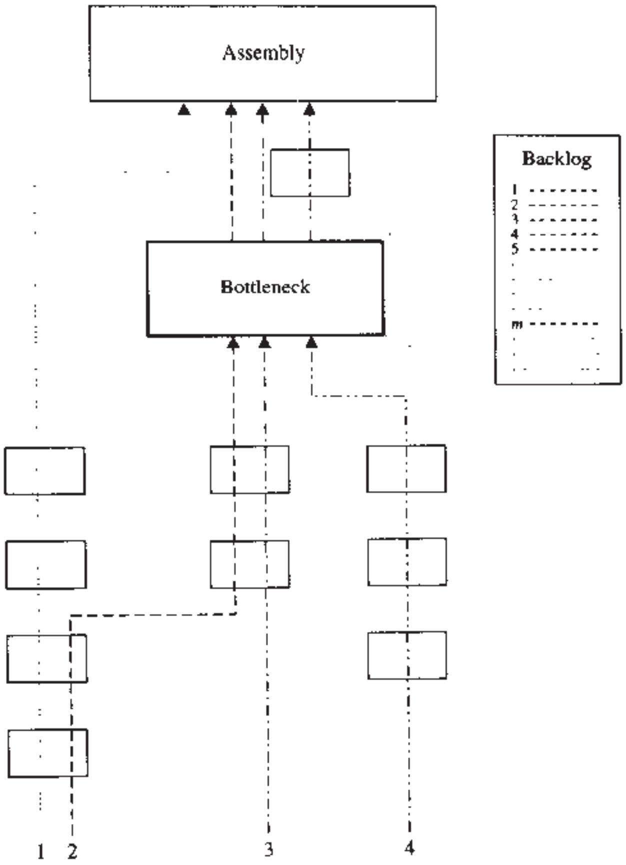
Consider the situation illustrated in Figure 14.13, which shows four distinct product routings, three of which pass through the bottleneck. The goal of a PFB strategy is to ensure that jobs are released so that they arrive at the bottleneck a specified time before they are needed (i.e., so that waiting jobs will form a buffer in front of the bottleneck to prevent random variations from causing it to starve).

To make our approach precise, let

b_i = time required on bottleneck by job i on backlog. Note that jobs on different routings may have different processing times, and there may even be different families within same routing having different processing times.

ℓ_i = average time after release required for job i to reach bottleneck. Note that this time involves processing on nonbottleneck resources only. Since most

FIGURE 14.13
Routings in a job shop



of queueing will occur at bottleneck, this time should be relatively constant for a given routing.[5] However, these times may differ substantially across routings.

L = specified time for jobs to wait in buffer in front of bottleneck. This is a user-specified constant that depends on how much time protection is desired at bottleneck.

Now we can compute the amount of work at the bottleneck in the line by summing the b_i values. Suppose that the work backlog contains jobs in the sequence they will be worked on at the bottleneck, and suppose job 1 represents the current job being worked on at the bottleneck (where b_1 represents its remaining processing time). Then the amount of time until the bottleneck will be available to work on job j is

$$\sum_{i=1}^{j-1} b_i$$

Our goal is to release jobs on the backlog so that they will wait L time units in front of the bottleneck. Since job j takes ℓ_j on average to get to the bottleneck, we should

[5] This is in sharp contrast with MRP, which assumes constant lead times through the entire plant including the bottleneck. Because MRP does not maintain constant loadings on the plant, actual cycle times can vary greatly, making the constant-lead-time assumption very poor.

release job j whenever

$$\sum_{i=1}^{j-1} b_i \leq \ell_j + L$$

Therefore, if we track the number

$$\sum_{i=1}^{j-1} b_i - \ell_j - L \tag{14.1}$$

for every job on the backlog and release jobs when this quantity hits (or goes below) zero, we will maintain a constant workload on the bottleneck and jobs should arrive on average L time units before they are needed at the bottleneck. As long as L is large enough to prevent significant delays at the bottleneck, the actual work sequence at the bottleneck should be able to match the sequence on the work backlog reasonably well.

Notice that if the ℓ_j lead times differ for different routings, then the release sequence may be different from the sequence on the work backlog. All other things being equal, a job with a large ℓ_j will be released earlier than a job with a small ℓ_j, as one would expect, since its index in Equation (14.1) will go negative sooner. Furthermore, since the work backlog may have intervals during which no jobs along certain routings are required, this system may let WIP along some routings fall to zero at some points. Thus, while this mechanism induces a WIP cap it is not CONWIP in the sense of maintaining constant loadings along routings.

The PFB logic we have described so far is fine for routings 2, 3, and 4, but does not cover routing 1, which does not run through the bottleneck.[6] A sensible approach for this routing is to control it as a CONWIP loop. This will be effective as long as the need for parts from routing 1 is relatively stable. If the final assembly sequence contains intervals during which there is no need for routing 1 parts, then we might want to modify the CONWIP logic to include a requirement that the part be required within a certain time window (e.g., a week) before releasing it. Thus, we would release jobs when *both* the WIP level in routing 1 fell below its target level *and* the next part was needed within a specified time window.

14.4.3 Shop Floor Control and Scheduling

This last point about holding parts out until they are within a window of their due date makes it clear that there is potentially a strong link between the shop floor control module and the sequencing and scheduling module. If we have generated a schedule using the sequencing and scheduling module, then we can control individual routings by releasing jobs according to this schedule, *subject to a WIP cap*. That is, jobs will be released whenever the (capacity-adjusted) WIP along the routing is below the target level and a job is within a specified time window of its scheduled release date. If the schedule contains enough work to keep the routing fully loaded, this approach is equivalent to CONWIP. If there are gaps in the schedule for products along a routing, then the WIP level along that routing may fall below the target level, or even to zero.

A variety of scheduling systems could be used in conjunction with a WIP cap mechanism in this manner. We will discuss scheduling approaches based on the conveyor model that are particularly well suited to this purpose in Chapter 15. But one could also

[6] Observe that although routings 2 and 3 share nonbottleneck resources, we do not consider this in the release mechanism. As long as these shared resources are not close to being bottlenecks, this will probably work well. However, if these resources can become bottlenecks depending on the product mix, more complex scheduling and release methods may be required. We will discuss this in Chapter 15.

use something less ideal, such as MRP. The planned order releases generated by MRP represent a schedule. Instead of following these releases independently of what is going on in the factory, one could block releases along routings whose WIP levels are too high, and move up releases (up to a specified amount) along routings whose WIP levels are too low. The fixed-lead-time assumption of MRP will still tend to make the schedule inaccurate. But by forcing compliance with a WIP cap, this SFC approach will at least prevent the dreaded WIP explosion. The benefits of capping WIP in an MRP system were pointed out long ago in the MRP literature (Wight 1970), but mechanisms for actually achieving this have been rare in practice.

14.5 Production Tracking

As we mentioned, the SFC module is the point of contact with the real-time evolution of the plant. Therefore, it is the natural place to monitor plant behavior. We are interested in both the short term, where the concern is making schedule, and the long term, where the concern is collecting accurate data for planning purposes. Although individual plants may have a wide range of specific data requirements, we will restrict our attention to two generic issues: monitoring progress toward meeting our schedule in the short term, and tracking key capacity parameters for use in other planning modules in the long term.

14.5.1 Statistical Throughput Control

In the short term, the primary question concerns whether we are on track to make our scheduled commitments. If the line is running as a CONWIP loop with a specified production quota, then the question concerns whether we will make the quota by the end of the period (e.g., by the end of the day or week). If we are following a schedule for the routing, then this depends on whether we will be on schedule at the next overtime opportunity. If there is a good chance that we will be behind schedule, we may want to prepare for overtime (notify workers). Alternatively, if the SFC module can provide early enough warning that we are seriously behind schedule, we may be able to reallocate resources or take other corrective action to remedy the problem.

We can use techniques similar to those used in statistical process control (SPC) to answer the basic short-term production tracking questions. Because of the analogy with SPC, we refer to this function of the SFC module as **statistical throughput control (STC).** To see how STC works, we consider production in a CONWIP loop during a single production period. Common examples of periods are (1) an eight-hour shift (with a four-hour preventive maintenance period available for overtime), (2) first and second shifts (with third shift available for overtime), and (3) regular time on Monday through Friday (with Saturday and Sunday available for overtime).

We denote the beginning of the period as time 0 and the end of the regular time period as time R. At any intermediate point in time t, where $0 \leq t \leq R$, we must compare two pieces of information:

n_t = *actual* cumulative production by line, possibly in capacity-adjusted units, in time interval $[0, t]$

S_t = *scheduled* cumulative production for line for time interval $[0, t]$

First, note that since S_t represents *cumulative* scheduled production, it is always increasing in t. However, if we are measuring actual production at a point in the routing prior to an inspection point, at which yield fallout is possible, then n_t could potentially decrease. Second, note that if the line uses a detailed schedule, S_t may increase unevenly. However, if it uses a periodic production quota, without a detailed schedule, so that the target is to complete Q units of production by time R, then we assume that S_t is linear (i.e., constant) on the interval, so that

$$S_t = Q\frac{t}{R}$$

and hence $S_R = Q$. Figure 14.14 illustrates two possibilities for S_t.

Ideally, we would like actual production n_t to equal scheduled production S_t at every point in time between 0 and R. Of course, because of random variations in the plant, this will virtually never happen. Therefore, we are interested in characterizing how far ahead of or behind schedule we are. We could plot $n_t - S_t$ as a function of time t, to show this in units of production. When $n_t - S_t > 0$, we are ahead of schedule; when $n_t - S_t < 0$, we are behind it. However, the difference between n_t and S_t does not give direct information on how difficult it will be to make up a shortage or how much cushion is provided by an overage. Therefore, a more illuminating piece of information is the *probability of being on schedule by the end of the regular time period*, given how far we are ahead or behind now.

In Appendix 14A we derive an expression for this probability under the assumption that we can approximate the distribution of production during any interval of time by using the normal distribution. From a practical implementation standpoint, however, it is convenient to use the formula from Appendix 14A to precompute the overage levels (that is, $n_t - S_t$) that cause the probability of missing the quota to be any specified level α. If we know the mean and standard deviation of production during regular time (in capacity-adjusted units), denoted by μ and σ, this can be accomplished as follows.

Define x to be

$$x = -\frac{(\mu - Q)(R - t)}{R} - z_\alpha \sigma \sqrt{\frac{R - t}{R}} \tag{14.2}$$

where z_α is found from a standard normal table such that $\Phi(z_\alpha) = \alpha$. We show in Appendix 14A that if the overage level at time t is equal to x (that is, $n_t - S_t = x$), then the probability of missing the quota is exactly α. If $n_t - S_t > (<)\, x$, then the probability of missing quota is less than (greater than) α.

We can display this information in simple graphical form. Figure 14.15 plots the x values for specific probabilities of missing the quota. We have chosen to display these

FIGURE 14.14

Scheduled cumulative production functions, S_t

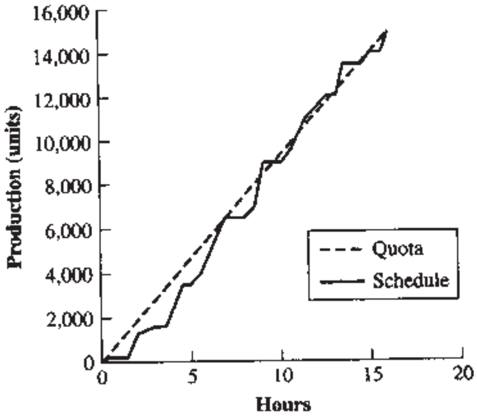
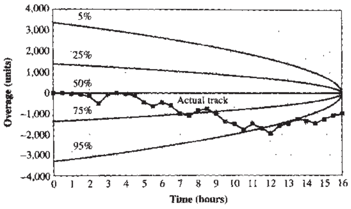


Figure 14.15

An STC chart when quota is equal to capacity



curves for probabilities of 5 percent, 25 percent, 50 percent, 75 percent, and 95 percent. In this example we are assuming a production quota, where regular time consists of two shifts, for a total of 16 hours, and historical data show that average production during 16 hours is 15,000 units and $\sigma = 2{,}000$ units. Quota is set equal to average capacity. That is, $S_t = Q_t/R$, where $Q = \mu = 15{,}000$. The curves in Figure 14.15 give an at-a-glance indication of how we stand relative to making the quota. For instance, if the overage level at time t (that is, $n_t - S_t$) lies exactly on the 75 percent curve, then the probability of missing the quota is 75 percent. On the basis of this information, the line manager may take action (e.g., shift workers) to speed things up. If $n_t - S_t$ rises above the 50 percent mark, this indicates that the action was successful. If it falls, say, below the 95 percent mark at time $t = 12$, then making the quota is getting increasingly improbable and perhaps it is time to announce overtime.

Notice that in Figure 14.15 the critical value (that is, x) for $\alpha = 0.5$ is always zero. The reason for this is that since the quota is set exactly equal to mean production, we always have a 50-50 chance of making it when we are exactly on time. The other critical values follow curved lines. For instance, the curve for $\alpha = 0.25$ indicates that we must be quite far ahead of scheduled production early in the regular time period to have only a 25 percent chance of missing the quota, but we must only be a little ahead of schedule near the end to have this same chance of missing the quota. The reason, of course, is that near the end of the period we do not have much of the quota remaining, and therefore less of a cushion is required to improve our chances of making it.

The Chapter 13 discussion on setting production quotas in pull systems pointed out that it may well be economically attractive to set the quota below mean regular time. When this is the case, we can still use Equation (14.2) to precompute the critical values for various probabilities of missing the quota. Figure 14.16 gives a graphical display of a case with a quota $Q = 14{,}000$ units, which is below mean regular time capacity $\mu = 15{,}000$ units. Notice that in this case, if we start out with no shortage or overage (that is, $n_0 - S_0 = 0$), then we begin with a greater than 50 percent chance of making the quota. This is because we have set the quota below the amount we can make on average during a regular time period. Since $Q < \mu$, on average we should be able to achieve a pace such that $n_t - S_t$ goes positive and continues to increase, that is, until the quota is reached and either production stops or we work ahead on the next period's quota. If something goes wrong, so that we fail to exceed the pace, then the position of the $n_t - S_t$ curve allows us to determine at a glance the probability of making the quota, given that we achieve historical average pace from time t until the end of regular time.

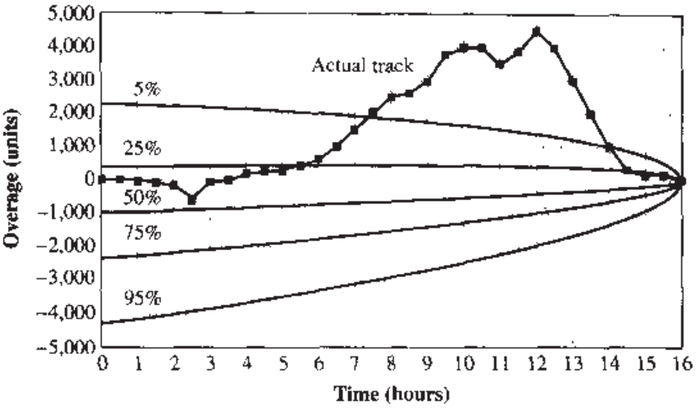


Figure 14.16

An STC chart when the quota is less than capacity

STC charts like those illustrated in Figures 14.15 and 14.16 can be generated by using Equation (14.2) and data on actual production (that is, n_t). The computer terminals of the CONWIP controller (see Figure 14.9) are a natural place to display these charts for CONWIP lines. STC charts can also be maintained and displayed at any critical resource in the plant.

STC charts can be useful even if n_t is not tracked in real time. For instance, if regular time consists of Monday through Friday and we only get readings on actual throughput at the end of each day, we could update the STC chart daily to indicate our chances for achieving the quota.

Finally, STC charts can be particularly useful at a critical resource that is shared by more than one routing. For instance, a system with two different circuit board lines running through a copper plating process could maintain separate STC charts for the two routings. Line managers could make decisions about which routing to work on from information about the quota status of the two routings. If line 1 is safely ahead of the quota, while line 2 is behind, then it makes sense to work on line 2 if incoming parts are available. Of course, we may need to use the information from the STC charts judiciously, to avoid rapid switches between lines if switching requires a significant setup.

14.5.2 Long-Range Capacity Tracking

In addition to providing short-term information to workers and managers, a production tracking system should provide input to other planning functions, such as aggregate and workforce planning and quota setting. The key data needed by these functions are the mean and standard deviation of regular time production of the plant in standard units of work. Since we are continually monitoring output via the SFC module, this is a reasonable place to collect this information.

In the following discussion, we assume that we can observe directly the amount of work (in capacity-adjusted standard units, if appropriate) completed during regular time. In a rigid quota system, in which work is stopped when the quota is achieved, even if this happens before the end of regular time, this procedure should *not* be used, since it will underestimate true regular time capacity. Instead, data should be collected on the mean and standard deviation of the *time to make quota*, which could be shorter or longer than the regular time period, and convert these to the mean and standard deviation of regular time production. The formulas for making this conversion are given in Spearman et al. (1989).

Since actual production during regular time is apt to fluctuate up and down due to random disturbances, it makes sense to smooth past data to produce estimates of the capacity parameters that are not inordinately sensitive to noise. The technique of exponential smoothing (Appendix 13A) is well suited to this task. We can use this method to take past observations of output to predict future capacity.

Let μ and σ represent the mean and standard deviation, respectively, of regular time production. These are the quantities we wish to estimate from past data. Let Y_n represent the nth observation of the amount produced during regular time, $\hat{\mu}(n)$ represent the nth smoothed estimate of regular time capacity, $\hat{T}(n)$ represent the nth smoothed trend, and α and β represent smoothing constants. We can iteratively compute $\hat{\mu}(n)$ and $\hat{T}(n)$ as

$$\hat{\mu}(n) = \alpha Y_n = (1 - \alpha)[\hat{\mu}(n - 1) + \hat{T}(n - 1)] \tag{14.3}$$

$$\hat{T}(n) = \beta[\hat{\mu}(n) - \hat{\mu}(n - 1)] + (1 - \beta)\hat{T}(n - 1) \tag{14.4}$$

At the end of each regular time period, we receive a new observation of output Y_n and can recompute our estimate of mean regular time capacity $\hat{\mu}(n)$. To start the method, we need estimates of $\hat{\mu}(0)$ and $\hat{T}(0)$. These can be reasonable guesses or statistical estimates based on historical data. Depending on the values of α and β, the effect of these initial values of $\hat{\mu}(0)$ and $\hat{T}(0)$ will "wash out" after a few actual observations.

Because we are making use of exponential smoothing with a trend, the system can also be used to chart improvement progress. The trend $\hat{T}(n)$ is a good indicator of capacity improvements. If positive, then average output is increasing. In a sell-all-you-can-make environment, higher mean capacity will justify higher production quotas and hence greater profits.

Recall that our computation of economic production quotas in Chapter 13 required the mean μ—*and* standard deviation σ—of regular time production. We can use exponential smoothing to track this parameter as well. Since variance is a much noisier statistic to track than the mean, it is more difficult to track trends explicitly. For this reason, we advocate using exponential smoothing with no trend.

Let Y_n represent the nth observation of the amount produced during regular time production, $\hat{\mu}(n)$ represent the nth estimate of mean regular time capacity, and γ denote a smoothing constant. Recall that the definition of variance of a random variable X is

$$\text{Var}(X) = E[(X - E[X])^2]$$

After the nth observation, we have estimated the mean of regular time capacity as $\hat{\mu}(n)$. Hence, we can make an estimate of the variance of regular time capacity after the nth observation as

$$\left[Y_n - \hat{\mu}(n)\right]^2$$

Since these estimates will be noisy, we smooth them with previous estimates to get

$$\hat{\sigma}^2(n) = \gamma\left[Y_n - \hat{\mu}(n)\right]^2 + (1 - \gamma)\hat{\sigma}^2(n - 1) \tag{14.5}$$

as our nth estimate of the variance of regular time production.

As usual with exponential smoothing, an estimate of $\hat{\sigma}^2(0)$ must be supplied to start the iteration. Thereafter, each new observation of regular time output yields a new estimate of the variance of regular time production. As we observed in Chapter 13, smaller variance enables us to set the quota closer to mean capacity and thereby yields greater profit. Therefore, a downward trend in $\hat{\sigma}^2(n)$ is a useful measure of an improving production system.

We now illustrate these calculations by means of the example described in Table 14.1. Regular time periods consist of Monday through Friday (two shifts per day), and we have collected 20 weeks of past data on weekly output. As a rough starting point we optimistically estimate capacity at 2,000 units per week, so we set $\hat{\mu}(0) = 2{,}000$. We have no evidence of a trend, so we set $\hat{T}(0) = 0$. We make a guess that the standard deviation of regular time production is around 100, so we set $\hat{\sigma}^2(0) = 100^2 = 10{,}000$. We will choose our smoothing constants to be

$$\begin{aligned} \alpha &= 0.5 \\ \beta &= 0.2 \\ \lambda &= 0.4 \end{aligned}$$

Of course, as we discussed in Appendix 13A, choosing smoothing constants is something of an art, so trial and error on past data may be required to obtain reasonable values in actual practice.

Now we can start the smoothing process. Regular time production during the first period is 1,400 units, so using Equation (14.3), we compute our smoothed estimate of mean regular time capacity as

$$\begin{aligned} \hat{\mu}(1) &= \alpha Y_1 + (1-\alpha)[\hat{\mu}(0) + \hat{T}(0)] \\ &= 0.5(1{,}400) + (1-0.5)(2{,}000+0) \\ &= 1{,}700 \end{aligned}$$

TABLE 14.1 Exponential Smoothing of Capacity Parameters

n	Y_n	$\hat{\mu}(n)$	$\hat{T}(n)$	$\hat{\sigma}^2(n)$	$\hat{\sigma}(n)$
0	—	2,000.0	0.0	10,000.0	100.0
1	1,400	1,700.0	−60.0	42,000.0	204.9
2	1,302	1,471.0	−93.8	36,624.4	191.4
3	1,600	1,488.6	−71.5	26,938.6	164.1
4	2,100	1,758.5	−3.2	62,801.1	250.6
5	1,800	1,777.7	1.2	37,880.4	194.6
6	2,150	1,964.4	38.4	36,500.0	191.0
7	2,450	2,226.4	83.1	41,898.8	204.7
8	2,200	2,254.7	72.1	26,337.7	162.3
9	2,600	2,463.4	99.4	23,263.2	152.5
10	2,100	2,331.4	53.2	35,382.6	188.1
11	2,200	2,292.3	34.7	24,636.7	157.0
12	2,600	2,463.5	62.0	22,235.7	149.1
13	2,800	2,662.7	89.4	20,877.2	144.5
14	2,300	2,526.1	44.2	32,973.8	181.6
15	2,900	2,735.2	77.2	30,653.1	175.1
16	2,800	2,806.2	76.0	18,407.1	135.7
17	2,650	2,766.1	52.7	16,433.0	128.2
18	3,000	2,909.4	70.9	13,142.7	114.6
19	2,750	2,865.1	47.8	13,188.1	114.8
20	3,150	3,031.5	71.5	13,531.0	116.3

Similarly, we use Equation (14.4) to compute the smoothed trend as

$$
\begin{aligned}
\hat{T}(1) &= \beta[\hat{\mu}(1) - \hat{\mu}(0)] + (1 - \beta)\hat{T}(0) \\
&= 0.2(1{,}700 - 2{,}000) + (1 - 0.2)(0) \\
&= -60
\end{aligned}
$$

Finally, we use Equation (14.5) to compute the smoothed estimate of variance of regular time production as

$$
\begin{aligned}
\hat{\sigma}^2(1) &= \gamma[Y_n - \hat{\mu}(1)]^2 + (1 - \gamma)\hat{\sigma}^2(0) \\
&= 0.4(1{,}400 - 1{,}700)^2 = (1 - 0.4)(10{,}000) \\
&= 42{,}000
\end{aligned}
$$

Thus, the smoothed estimate of standard deviation of regular time production is $\hat{\sigma}(1) = \sqrt{42{,}000} = 204.9$.

We can continue in this manner to generate the numbers in Table 14.1. A convenient way to examine these data is to plot them graphically. Figure 14.17 compares the smoothed estimates with the actual values of regular time production. Notice that the smoothed estimate follows the upward trend of the data, but with less variability from period to period (it is called *smoothing*, after all). Furthermore, this graph makes it apparent that our initial estimate of regular time capacity of 2,000 units per week was somewhat high. To compensate, the smoothed estimate trends downward for the first few periods, until the actual upward trend forces it up again.

These trends can be directly observed in Figure 14.18, which plots the smoothed trend after each period. Because of the high initial estimate of $\hat{\mu}(0)$, this trend is initially negative. The eventual positive trend indicates that capacity is increasing in this plant, a sign that improvements are having an effect on the operation.

Finally, Figure 14.19 plots the smoothed estimate of the standard deviation of regular time production. This estimate appears to be constant or slightly decreasing. A decreasing estimate is an indication that plant improvements are reducing variability in output. Both this and the smoothed trend provide us with hard measures of continual improvement.

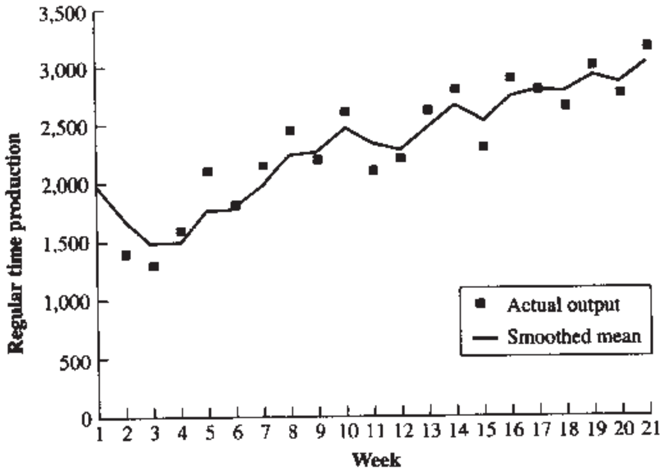


FIGURE 14.17

Exponential smoothing of mean regular time capacity

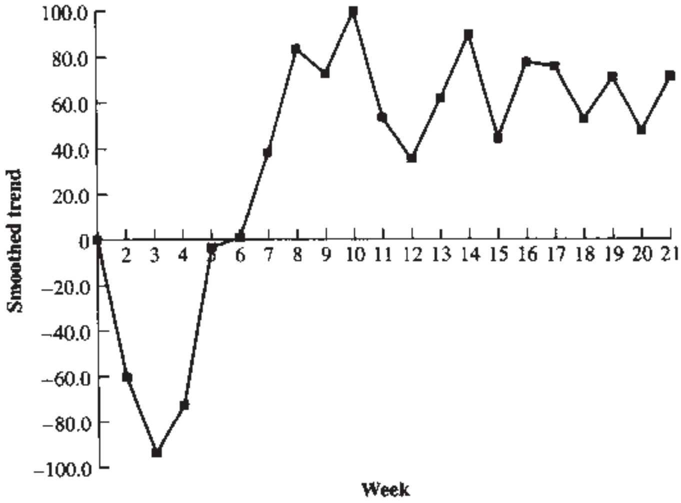


FIGURE 14.18

Exponential smoothing of trend in mean regular time capacity

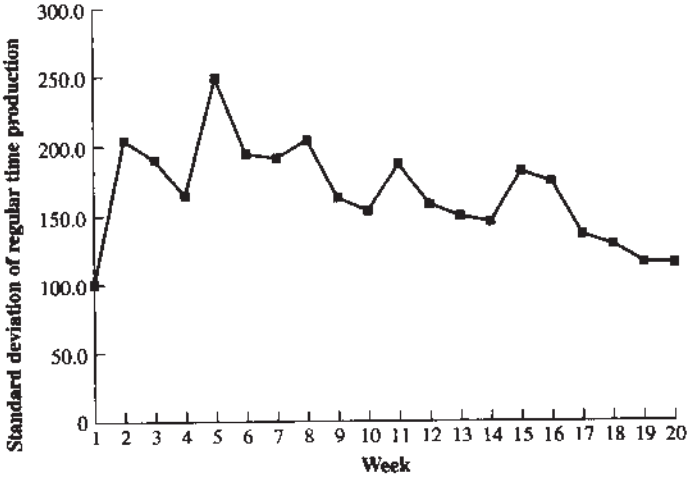


FIGURE 14.19

Exponential smoothing of variance of regular time capacity

14.6 Conclusions

In this chapter, we have spent a good deal of time discussing the shop floor control (SFC) module of a production planning and control (PPC) system. We have stressed that a good SFC module can do a great deal more than simply govern the movement of material into and through the factory. As the lowest-level point of contact with the manufacturing process, SFC plays an important role in shaping the management problems that must be faced. A well-designed SFC module will establish a predictable, robust system with controls whose complexity is appropriate for the system's needs.

Because manufacturing systems are different, a uniform SFC module for all applications is impractical, if not impossible. A module that is sufficiently general to handle a broad range of situations is apt to be cumbersome for simple systems and ill suited for specific complex systems. More than any other module in the PPC hierarchy, the SFC module is a candidate for customization. It may make sense to make use of commercial bar coding, optical scanning. local area networks, statistical process control, and other technologies as components of an SFC module. However, there is no substitute for careful integration done with the capabilities and needs of the system in mind. It is our hope that the manufacturing professionals reading this book will provide such integration, using the basics, intuition, and synthesis skills they have acquired here and elsewhere.

Since we do not believe it is possible to provide a cookbook scheme for devising a suitable SFC module, we have taken the approach of starting with simple systems, highlighting key issues, and extending our approach to various more complex issues. Our basic scheme is to start with a simple set of CONWIP lines as the incumbent and ask why such a setup would not work. If it does work, as we believe it can in relatively uncomplicated flow shops, then this is the simplest, most robust solution. If not, then more complex schemes, such as that of pull-from-bottleneck (PFB), may be necessary. We hope that the variations on CONWIP we have offered are sufficient to spur the reader to think creatively of options for specific situations beyond those discussed here.

One last issue we have emphasized is that feedback is an *essential* feature of an effective production planning and control system. Unfortunately, many PPC systems evolve in a distributed fashion, with different groups responsible for different facets of the planning process. The result is that inconsistent data are used, communication between decision makers breaks down, and factionalism and finger pointing, instead of cooperation and coordination, become the standard response to problems. Furthermore, without a feedback mechanism, overly optimistic data (e.g., unrealistically high estimates of capacity) can persist in planning systems, causing them to be untrustworthy at best and downright humorous at worst. Statistical throughput control is one explicit mechanism for forcing needed feedback with regard to capacity data. Similar approaches can be devised to promote feedback on other key data, such as process yields, rework frequency, and learning curves for new products. The key is for management to be sensitive to the potential for inconsistency and to strive to make feedback systemic to the PPC hierarchy. Furthermore, to be effective, feedback mechanisms must be used in a spirit of problem solving, not one of blame fixing.

Although the SFC module performs some of the most lowly and mundane tasks in a manufacturing plant, it can play a critical role in the overall effectiveness of the system. A well-designed SFC module establishes a predictable environment upon which to build the rest of the planning hierarchy. Appropriate feedback mechanisms can collect useful data for such planning and can promote an environment of ongoing improvement. To recall our quote from the beginning of this chapter,

> *Even a journey of one thousand li begins with a single step.*
>
> Lao Tzu

The SFC module is not only the first step toward an effective production planning and control system, it is a very important step indeed.

Appendix 14A
Statistical Throughput Control

The basic quantity needed to address several short-term production tracking questions is the probability of making the quota by the end of regular time production, given that we know how much has been produced thus far. Since output from each line must be recorded in order to maintain a constant WIP level in the line, a CONWIP line will have the requisite data on hand to make this calculation.

To do this, we define the length of regular time production as R. We assume that production during this time, denoted by N_R, is normally distributed, with mean μ and standard deviation σ. We let N_t represent production, in standard units, during $[0, t]$, where $t \leq R$. We model N_t as continuous and normally distributed with mean $\mu t/R$ and variance $\sigma^2 t/R$. In general, the assumption that production is normal will often be good for all but small values of t. The assumption that the mean and variance of N_t are as given here is equivalent to assuming that production during nonoverlapping intervals is independent. Again, this is probably a good assumption except for very short intervals.

We are interested primarily in the process $N_t - S_t$, where S_t is the cumulative scheduled production up to time t. If we are using a periodic production quota, then $S_t = Qt/R$. The quantity $N_t - S_t$ represents the overage, or amount by which we are ahead of schedule, at time t. If this quantity is positive, we are ahead; if negative, we are behind. In an ideal system with constant production rates, this quantity would always be zero. In a real system, it will fluctuate, becoming positive and/or negative.

From our assumptions, it follows that $N_t - Qt/R$ is normally distributed with mean $(\mu - Q)t/R$ and variance $\sigma^2 t/R$. Likewise, N_{R-t} is normally distributed, with mean $\mu(R-t)/R$ and variance $\sigma^2(R-t)/R$. Hence, if at time t, $N_t = n_t$, where $n_t - Qt/R = x$ (we are x units ahead of schedule), then we will miss the quota by time R only if $N_{R-t} < Q - n_t$. Thus, the probability of missing the quota by time R given a current overage of x is given by

$$\begin{aligned} P(N_{R-t} \leq Q - n_t) &= P\left(N_{R-t} \leq Q - x - \frac{Qt}{R}\right) \\ &= P\left(N_{R-t} \leq \frac{Q(R-t)}{R} - x\right) \\ &= \Phi\left[\frac{(Q-\mu)(R-t)/R - x}{\sigma\sqrt{(R-t)/R}}\right] \end{aligned}$$

where $\Phi(\cdot)$ represents the standard normal distribution.

From a practical implementation standpoint, it is more convenient to precompute the overage levels that cause the probability of missing the quota to be any specified level α. These can be computed as follows:

$$\Phi\left[\frac{(Q-\mu)(R-t)/R - x}{\sigma\sqrt{(R-t)/R}}\right] = \alpha$$

which yields

$$x = -\frac{(\mu - Q)(R-t)}{R} - z_\alpha \sigma \sqrt{\frac{R-t}{R}}$$

where z_α is chosen such that $\Phi(z_\alpha) = \alpha$. This x is the overage at time t that results in a probability of missing the quota exactly equal to α, and is Equation (14.2), upon which our STC charts are based.

Study Questions

1. What is the motivation for limiting the span of control of a manager to a specified number of subordinates or manufacturing processes? What problems might this cause in coordinating the plant?
2. We have repeatedly mentioned that throughput is an increasing function of WIP. Therefore, we could conceivably vary the WIP level as a way of matching production to the demand rate. Why might this be a poor strategy in practice?

3. What factors might make kanban inappropriate for controlling material flow through a job shop, that is, a system with many, possibly changing, routings with fluctuating volumes?
4. Why might we want to violate the WIP cap imposed by CONWIP and run a card deficit when a machine downstream from the bottleneck fails? If we allow this, what additional discipline might we want to impose to prevent WIP explosions?
5. What are the advantages ot breaking a long production line into tandem CONWIP loops? What are the disadvantages?
6. For each of the following situations, indicate whether you would be inclined to use CONWIP (C), kanban (K), PFB (P), or an individual system (I) for shop floor control.
   a. A flow line with a single-product family.
   b. A paced assembly line fed from inventory storage.
   c. A steel mill where casters feed hot strip mills (with slab storage in between), which feed cold rolling mills (with coil storage in between).
   d. A plant with several routings sharing some resources with significant setup times, and all routings are steadily loaded over time.
   e. A plant with many routings sharing some resources but where some routings are sporadically used.
7. What is meant by statistical throughput control, and how does it differ from statistical process control? Could one use SPC tools (i.e., control charts) for throughput tracking?
8. Why is the STC chart in Figure 14.15 symmetric, while the one in Figure 14.16 is asymmetric? What does this indicate about the effect of setting production quotas at or near average capacity?
9. Why might it make sense to use exponential smoothing with a linear trend to track mean capacity of a line? How could we judge whether exponential smoothing without a linear trend might work as well or better?
10. What uses are there for tracking the standard deviation of periodic output from a production line?

Problems

1. A circuit board manufacturing line contains an expose operation consisting of five parallel machines inside a clean room. Because of limited space, there is only room for five carts of WIP (boards) to buffer expose against upstream variability. Expose is fed by a coater line, which consists of a conveyor that loads boards at a rate of three per minute and requires roughly one hour to traverse (i.e., a job of 60 boards will require 20 minutes to load plus one hour for the last loaded board to arrive in the clean room at expose). Expose machines take roughly two hours to process jobs of 60 boards each. Current policy is that whenever the WIP inside the clean room reaches five jobs (in addition to the five jobs being worked on at the expose machines), the coater line is shut down for three hours. Both expose and the coater are subject to variability due to machine failures, materials shortages, operator unavailability, and so forth. When all this is factored into a capacity analysis, expose seems to be the bottleneck of the entire line.
   a. What problem might the current policy for controlling the coater present?
   b. What alternative would you suggest? Remember that expose is isolated from the rest of the line by virtue of being in a clean room and that because of this, the expose operators cannot see the beginning of the coater; nor can the coater loader easily see what is going on inside the clean room.
   c. How would your recommendation change if the capacity of expose were increased (say, by using floating labor to work through lunches) so that it was no longer the long-term bottleneck?
2. Consider a five-station line that processes two products, A and B. Station 3 is the bottleneck for both products. However, product A requires one hour per unit at the bottleneck, while

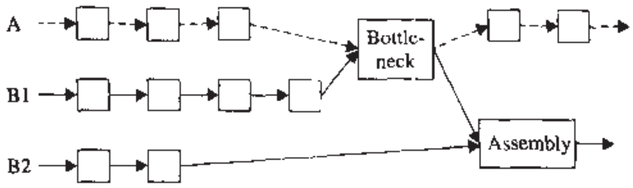


FIGURE 14.20
Pull-from-bottleneck production system

product B requires one-half hour. A modified CONWIP control policy is used under which the complexity-adjusted WIP is measured as the number of hours of work at the bottleneck. Hence, one unit of A counts as one unit of complexity-adjusted WIP, while one unit of B counts as one-half unit of complexity-adjusted WIP. The policy is to release the next job in the sequence whenever the complexity-adjusted WIP level falls to 10 or less.

a. Suppose the release sequence alternates between product A and B (that is, A-B-A-B-A-B-...). What will happen to the numbers of type A and type B jobs in the system over time?

b. Suppose the release sequence alternates between 10 units of A and 10 units of B. Now what happens to the numbers of type A and type B jobs in the system over time?

c. The JIT literature advocates a sequence like the one in *a*. Why? Why might some lines need to make use of a sequence like the one in *b*?

3. Consider the two-product system illustrated in Figure 14.20. Product A and component 1 of product B pass through the bottleneck operation. Components 1 and 2 of product B are assembled at the assembly operation. Type A jobs require one hour of processing at the bottleneck, while type B jobs require one and one-half hours. The lead time for type A jobs to reach the bottleneck from their release point is two hours. Component 1 of type B jobs takes four and one-half hours to react the bottleneck. The sequence of the next eight jobs to be processed at the bottleneck is as follows:

Job index	1	2	3	4	5	6	7	8
Job type	A	A	B	B	B	B	A	B

Jobs 1 through 6 have already been released but have not yet been completed at the bottleneck. Suppose that the system is controlled using the pull-from-the-bottleneck method described in Section 14.4.2, where the planned time at the bottleneck is $L = 4$ hours.

a. When should job 7 be released (i.e., now or after the completion of that job currently in the system)?

b. When should job 8 be released (i.e., now or after the completion of that job currently in the system)? Are jobs necessarily released in the order they will be processed at the bottleneck? Why or why not?

c. If we only check to see whether new jobs should be released when jobs are completed at the bottleneck, will jobs wait at the bottleneck more than, less than, or equal to the target time L? (*Hint:* What is the expected waiting time of job 8 at the bottleneck?) Could these be cases in which we would want to update the current workload at the bottleneck more frequently than at completion times of jobs?

d. Suppose that the lead time for component 2 of product B to reach assembly is one hour. If we want component 2 to wait for one and one-half hours on average at assembly, when should it be released relative to its corresponding component 1?

4. Consider a line that builds toasters runs five days per week, one shift per day (or 40 hours per week). A periodic quota of 2,500 toasters has been set. If this quota is not met by the end of work on Friday, overtime on the weekend is run to make up the difference. Historical data indicate that the capacity of the line is 2,800 toasters per week, with a standard deviation of 300 toasters.

a. Suppose at hour 20 we have completed 1,000 toasters. Using the STC model, estimate the probability that the line will be able to make the quota by the end of the week.

b. How many toasters must be completed by hour 20 to ensure a probability of 0.9 of making the quota?

c. If the weekly quota is increased to 2,800 toasters per week, how does the answer to *b* change?

5. Output from the assembly line of a farm machinery manufacturer that produces combines has been as follows for the past 20 weeks:

Week Output	1 22	2 21	3 24	4 30	5 25	6 25	7 33	8 40	9 36	10 39
Week Output	11 50	12 55	13 44	14 48	15 55	16 47	17 61	18 58	19 55	20 60

a. Use exponential smoothing with a linear trend and smoothing constants $\alpha = 0.4$ and $\beta = 0.2$ to track weekly output for weeks 2 to 20. Does there appear to be a positive trend to the data?

b. Using mean square deviation (MSD) as your accuracy measure, can you find values of α and β that fit these data better than those given in *a*?

c. Use exponential smoothing (without a linear trend) and a smoothing constant $\gamma = 0.2$ to track variance of weekly output for weeks 2 to 20. Does the variance seem to be increasing, decreasing, or constant?

CHAPTER

15 Production Scheduling

Let all things be done decently and in order.
I Corinthians

15.1 Goals of Production Scheduling

Virtually all manufacturing managers want on-time delivery, minimal work in process, short customer lead times, and maximum utilization of resources. Unfortunately, these goals conflict. It is much easier to finish jobs on time if resource utilization is low. Customer lead times can be made essentially zero if an enormous inventory is maintained. And so on. The goal of production scheduling is to strike a profitable balance among these conflicting objectives.

In this chapter we discuss various approaches to the scheduling problem. We begin with the standard measures used in scheduling and a review of traditional scheduling approaches. We then discuss why scheduling problems are so hard to solve and what implications this has for real-world systems. Next we develop practical scheduling approaches, first for the bottleneck resource and then for the entire plant. Finally, we discuss how to interface scheduling—which is push in concept—with a pull environment such as CONWIP.

15.1.1 Meeting Due Dates

A basic goal of production scheduling is to meet due dates. These typically come from one of two sources: directly from the customer or in the form of material requirements for other manufacturing processes.

In a make-to-order environment, customer due dates drive all other due dates. As we saw in Chapter 3, a set of customer requirements can be exploded according to the associated bills of material to generate the requirements for all lower-level parts and components.

In a make-to-stock environment there are no customer due dates, since all customer orders are expected to be filled immediately upon demand. Nevertheless, at some point, falling inventory triggers a demand on the manufacturing system. Demands generated in this fashion are just as real as actual customer orders since, if they are not met, customer demands will eventually go unfilled. These stock replenishment demands are exploded into demands for lower-level components in the same fashion as customer demands.

Several measures can be used to gauge due date performance, including these:

Service level (also known as simply **service**), typically used in make-to-order systems, is the fraction of orders filled on or before their due dates. Equivalently, it is the fraction of jobs whose cycle time is less than or equal to the planned lead time.

Fill rate is the make-to-stock equivalent of service level and is defined as the fraction of demands that are met from inventory, that is, without backorder.

Lateness is the difference between the order due date and the completion date. If we define d_j as the due date and c_j as the completion time of job j, the lateness of job j is given by $L_j = c_j - d_j$. Notice that lateness can be positive (indicating a late job) or negative (indicating an early job). Consequently, small *average* lateness has little meaning. It could mean that all jobs finished near their due dates, which is good; or it could mean that for every job that was very late there was one that was very early, which is bad. For lateness to be a useful measure, we must consider its *variance* as well as its *mean*. A small mean and variance of lateness indicates that most jobs finish on or near their due dates.

Tardiness is defined as the lateness of a job if it is late and zero otherwise. Thus, early jobs have zero tardiness. Consequently, *average* tardiness *is* a meaningful measure of customer due date performance.

These measures suggest several objectives that can be used to formulate scheduling problems. One that has become classic is to "minimize average tardiness." Of course, it is classic only in the production scheduling research literature, not in industry. As one might expect, "minimize lateness variance" has also seen very little use in industry.

Service level and fill rate *are* used in industry. This is probably because tardiness is difficult to track and because the measures of average tardiness and lateness variance are not intuitive. The percentage of on-time jobs is simpler to state than something like "the average number of days late, with early jobs counting as zero" or "the standard deviation of the difference between job due date and job completion date." However, service level and fill rate have obvious problems. Once a job is late, it counts against service *no matter how late it is.* Naive approaches can thus lead to ridiculous schedules that call for such things as never finishing late jobs or lying to customers. We present a due date quoting procedure in Section 15.3.2 that avoids these difficulties.

15.1.2 Maximizing Utilization

In industry, cost accounting encourages high machine utilization. Higher utilization of capital equipment means higher return on investment, provided of course that the equipment is utilized to increase revenue (i.e., to create products that are in demand). Otherwise, high utilization merely serves to increase inventory, not profits. High utilization makes the most sense when producing a commodity item to stock.

Factory physics also promotes high utilization, provided cycle times, quality, and service are not degraded excessively. However, recall that the Capacity Law implies that 100 percent utilization is impossible. How close to full utilization a line can run and still have reasonable WIP and cycle time depends on the level of variability. The more variability a line has, the lower utilization must be to compensate. Furthermore, as the practical worst case in Chapter 7 illustrated, *balanced* lines have more congestion than unbalanced ones, especially when variability is high. This implies that it may well be attractive not to have near 100 percent utilization of *all* resources in the line.

A measure that is closely related to utilization is **makespan,** which is defined as the time it takes to finish a fixed number of jobs. For this set of jobs, the production rate is the number of jobs divided by the makespan, and the utilization is the production rate divided by the capacity. Although makespan is not widely used in industry, it has seen frequent use in the theoretical scheduling research.

The decision of what target to use for utilization is a strategic one that belongs at the top of the in-plant planning hierarchy (Chapter 13). Because high-level decisions are made less frequently than low-level ones, utilization cannot be adjusted to facilitate production scheduling. Similarly, the level of variability in the line is a consequence of high-level decisions (e.g., capacity and process design decisions) that are also made much less frequently than are scheduling decisions. Thus, for the purposes of scheduling we can assume that utilization targets and variability levels are given. In most cases, the target utilization of the bottleneck resource will be high. The one important exception to this is a highly variable and customized demand process requiring an extremely quick response time (e.g., ambulances and fire engines). Such systems typically have very low utilization and are not well suited to scheduling. We will assume throughout, therefore, that the system is such that a fairly high bottleneck utilization is desirable.

15.1.3 Reducing WIP and Cycle Times

As we discussed in Part II, there are several motives for keeping cycle times short, including these:

1. *Better responsiveness to the customer.* If it takes less time to make a product, the lead time to the customer can be shortened.
2. *Maintaining flexibility.* Changing the list (backlog) of parts that are planned to start next is less disruptive than trying to change the set of jobs already in process. Since shorter cycle times allow for later releases, they enhance this type of flexibility.
3. *Improving quality.* Long cycle times typically imply long queues in the system, which in turn imply long delays between defect *creation* and defect *detection.* For this reason, short cycle times support good quality.
4. *Relying less on forecasts.* If cycle times are longer than customers are willing to wait, production must be done in anticipation of demand rather than in response to it. Given the lack of accuracy of most demand forecasts, it is extremely important to keep cycle times shorter than quoted lead times, whenever possible.
5. *Making better forecasts.* The more cycle times exceed customer lead times, the farther out the forecast must extend. Hence, even if cycle times cannot be reduced to the point where dependence on forecasting is eliminated, cycle time reduction can shorten the forecasting time horizon. This can greatly reduce forecasting errors.

Little's Law (CT = WIP/TH) implies that reducing cycle time and reducing WIP are equivalent, provided that throughput remains constant. However, the Variability Buffering Law implies that reducing WIP without reducing variability will cause throughput to decrease. Thus variability reduction is generally an important component of WIP and cycle time reduction programs.

Although WIP and cycle time may be virtually equivalent from a reduction policy standpoint, they are not equivalent from a measurement standpoint. WIP is often easier to measure, since one can count jobs, while cycle times require clocking jobs in and out of the system. Cycle times become even harder to measure in assembly operations. Consider an automobile, for instance. Does the cycle time start with the ordering of the components such as spark plugs and steel, or when the chassis starts down the assembly line? In such cases, it is more practical to use Little's Law to obtain an indirect measure of cycle time by measuring WIP (in dollars) over the system under consideration and dividing by throughput (in dollars per day).

15.2 Review of Scheduling Research

Scheduling as a practice is as old as manufacturing itself. Scheduling as a research discipline dates back to the scientific management movement in the early 1900s. But serious analysis of scheduling problems did not begin until the advent of the computer in the 1950s and 1960s. In this section, we review key results from the theory of scheduling.

15.2.1 MRP, MRP II, and ERP

As we discussed in Chapter 3, MRP was one of the earliest applications of computers to scheduling. However, the simplistic model of MRP undermines its effectiveness. The reasons, which we noted in Chapter 5, are as follows:

1. MRP assumes that lead times are attributes of parts, independent of the status of the shop. In essence, MRP assumes infinite capacity.
2. Since MRP uses only one lead time for offsetting and since late jobs are typically worse than excess inventory, there is strong incentive to inflate lead times in the system. This results in earlier releases, larger queues, and hence longer cycle times.

As we discussed in Part II, these problems prompted some scheduling researchers and practitioners to turn to enhancements in the form of MRP II and, more recently, ERP. Others rejected MRP altogether in favor of JIT. However, the majority of scheduling researchers focused on mathematical formulations in the field of operations research, as we discuss next.

15.2.2 Classic Scheduling

We refer to the set of problems in this section as *classic* scheduling problems because of their traditional role as targets of study in the operations research literature. For the most part, these problems are highly simplified and generic, which has limited their direct applicability to real situations. However, despite the fact that they are not classic from an applications perspective, they can offer some useful insights.

Most classical scheduling problems address one, two, or possibly three machines. Other common simplifying assumptions include these:

1. All jobs are available at the start of the problem (i.e., no jobs arrive after processing begins).
2. Process times are deterministic.
3. Process times do not depend on the schedule (i.e., there are no setups).
4. Machines never break down.
5. There is no preemption (i.e., once a job starts processing, it must finish).
6. There is no cancellation of jobs.

These assumptions serve to reduce the scheduling problem to manageable proportions, in some cases. One reason is that they allow us to restrict attention to simplified schedules, called sequences. In general, a **schedule** gives the anticipated start times of each job on each resource, while a **sequence** gives only the order in which the jobs are to be done. In some cases, such as the single-machine problem with all jobs available when processing begins, a simple sequence is sufficient. In more complex problems, separate sequences for different resources may be required. And in some problems a full-blown schedule is necessary to impart the needed instructions to the system. Not surprisingly, the more complex the form of the schedule that is sought, the more difficult it is to find it.

Some of the best-known problems that have been studied in the context of the assumptions discussed in the operations research literature are the following.

> **Minimizing average cycle time on a single machine.** First, note that for the single-machine problem, the *total time* to complete all the jobs does not depend on the ordering—it is given by the sum of the processing times for the jobs. Hence an alternate criterion is needed. One candidate is the average cycle time (called **flow time** in the production scheduling literature), which can be shown to be minimized by processing jobs in order of their processing times, with the shortest job first and longest job last. This is called the **shortest process time (SPT)** sequencing rule. The primary insight from this result is that short jobs move through the shop more quickly than long jobs and therefore tend to reduce congestion.
>
> **Minimizing maximum lateness on a single machine.** Another possible criterion is the maximum lateness that any job is late, which can be shown to be minimized by ordering the jobs according to their due dates, with the earliest due date first and the latest due date last. This is called the **earliest due date (EDD)** sequencing rule. The intuition behind this approach is that if it is *possible* to finish all the jobs on time, EDD sequencing will do so.
>
> **Minimizing average tardiness on a single machine.** A third criterion for the single-machine problem is average tardiness. (Note that this is equivalent to total tardiness, since average tardiness is simply total tardiness divided by the number of jobs.) Unfortunately, there is no sequencing rule that is guaranteed to minimize this measure. Often EDD is a good heuristic, but its performance cannot be ensured, as we demonstrate in one of the exercises at the end of the chapter. Likewise, there is no sequencing rule that minimizes the variance of lateness. We will discuss the reasons why this scheduling problem and many others like it are particularly hard to solve.
>
> **Minimizing makespan on two machines.** When the production process consists of two machines, the total time to finish all the jobs, the makespan, is no longer fixed. This is because certain sequences might induce idle time on the second machine as it waits for the first machine to finish a job. Johnson (1954) proposed an intuitive algorithm for finding the sequence that minimizes makespan for this problem, which can be stated as follows: Separate the jobs into two sets, A and B. Jobs in set A are those whose process time on the first machine is less than or equal to the process time on the second machine. Set B contains the remaining jobs. Jobs in set A go first and in the order of the shortest process time first. Then jobs in set B are appended in order of the longest process time first. The result is a sequence that minimizes the makespan over the two machines.
>
> The insight behind Johnson's algorithm can be appreciated by noting that we want a short job in the first position because the second machine is idle until the first job finishes on the first machine. Similarly, we want a short job to be last

since the first machine is idle while the second machine is finishing the last job. Hence, the algorithm implies that small jobs are better for reducing cycle times and increasing utilization.

Minimizing makespan in job shops. The problem of minimizing the time to complete *n* jobs with general routings through *m* machines (subject to all the assumptions previously discussed) is a well-known hard problem in the operations research literature. The reason for its difficulty is that the number of possible schedules to consider is enormous. Even for the modestly sized 10-job, 10-machine problem there are almost 4×10^{65} possible schedules (more atoms than there are in the earth). Because of this a 10-by-10 problem was not solved optimally until 1988 by using a mainframe computer and five hour of computing time (Carlier and Pinson 1988).

A standard approach to this type of problem is known as **branch and bound.** The basic idea is to define a **branch** by selecting a partial schedule and define **bounds** by computing a lower limit on the makespan that can be achieved with a schedule that includes this partial schedule. If the bound on a branch exceeds the makespan of the best (complete) schedule found so far, it is no longer considered. This is a method of **implicit enumeration,** which allows the algorithm to consider only a small subset of the possible schedules. Unfortunately, even a very small fraction of these can be an incredibly large number, and so branch and bound can be tediously slow. Indeed, as we will discuss, there is a body of theory that indicates that any exact algorithm for hard problems, like the job shop scheduling problem, will be slow. This makes nonexact *heuristic* approaches a virtual necessity. We will list a few of the many possible approaches in our discussion of the complexity of scheduling problems.

15.2.3 Dispatching

Scheduling is hard, both theoretically (as we will see) and practically speaking. A traditional alternative to scheduling all the jobs on all the machines is to simply **dispatch**—sort according to a specified order—as they arrive at machines. The simplest dispatching rule (and also the one that seems fairest when dealing with customers) is **first-in, first-out (FIFO).** The FIFO rule simply processes jobs in the order in which they arrive at a machine. However, simulation studies have shown that this rule tends not to work well in complex job shops. Alternatives that can work better are the SPT or EDD rules, which we discussed previously. In fact, these are often used in practice, as we noted in Chapter 3 in our discussion of shop floor control in ERP. Literally hundreds of different dispatching rules have been proposed by researchers as well as practitioners (see Blackstone, et al. 1982 for a survey).

All dispatching rules, however, are *myopic* in nature. By their very definition they consider only local and current conditions. Since the best choice of what to work on now at a given machine depends on future jobs as well as other machines, we cannot expect dispatching rules to work well all the time, and, in fact, they do not. But because the options for scheduling realistic systems are still very limited, dispatching continues to find extensive use in industry.

15.2.4 Why Scheduling Is Hard

We have noted several times that scheduling problems are hard. A branch of mathematics known as **computational complexity analysis** gives a formal means for evaluating just how hard they are. Although the mathematics of computational complexity is beyond our scope, we give a qualitative treatment of this topic in order to develop an appreciation of why some scheduling problems cannot be solved optimally. In these cases, we are forced to go from seeking the *best* solution to finding a *good* solution.

Problem Classes. Mathematical problems can be divided into the following two classes according to their complexity:

1. **Class P problems** are problems that can be solved by algorithms whose computational time grows as a polynomial function of problem size.
2. **NP-hard problems** are problems for which there is no known polynomial algorithm, so that the time to find a solution grows exponentially (i.e., much more rapidly than a polynomial function) in problem size. Although it has not been definitively proved that there are no clever polynomial algorithms for solving NP-hard problems, many eminent mathematicians have tried and failed. At present, the preponderance of evidence indicates that efficient (polynomial) algorithms cannot be found for these problems.

Roughly speaking, class P problems are easy, while NP-hard problems are hard. Moreover, some NP-hard problems appear to be harder than others. For some, efficient algorithms have been shown empirically to produce good approximate solutions. Other NP-hard problems, including many scheduling problems, are even difficult to solve approximately with efficient algorithms.

To get a feel for what the technical terms **polynomial** and **exponential** mean, consider the single-machine sequencing problem with three jobs. How many ways are there to sequence three jobs? Any one of the three could be in the first position, which leaves two candidates for the second position, and only one for the last position. Therefore, the number of sequences or *permutations* is $3 \times 2 \times 1 = 6$. We write this as 3! and say "3 factorial." If we were looking for the best sequence with regard to some objective function for this problem, we would have to consider (explicitly or implicitly) six alternatives. Since the factorial function exhibits exponential growth, the number of alternatives we must search through, and therefore the amount of time required to find the optimal solution, also grows exponentially in problem size.

The reason this is important is that *any* polynomial function will *eventually* become dominated by *any* exponential function. For instance, the function $10{,}000n^{10}$ is a big polynomial, while the function $e^n/10{,}000$ appears small. Indeed, for small values of n, the polynomial function dominates the exponential. But at around $n = 60$ the exponential begins to dominate and by $n = 80$ has grown to be 50 million times larger than the polynomial function.

Returning to the single-machine problem with three jobs, we note that 3! does not seem very large. However, observe how quickly this function blows up: $3! = 6$, $4! = 24$, $5! = 120$, $6! = 720$, and so on. As the number of jobs to be sequenced becomes large, the number of possible sequences becomes quite ominous: $10! = 3{,}628{,}800$, $13! = 6{,}227{,}020{,}800$, and

$$25! = 15{,}511{,}210{,}043{,}330{,}985{,}984{,}000{,}000$$

To get an idea of how big this number is, we compare it to the national debt, which at the time of this writing had not yet reached $5 trillion. Nonetheless, suppose it were $5 trillion and we wanted to pay it in pennies. The 500 trillion pennies would cover almost one-quarter of the state of Texas. In comparison, 25! pennies would cover the *entire*

TABLE 15.1 Computer Times for Job Sequencing on a Slow Computer

Number of Jobs	Computer Time
5	0.12 millisec
6	0.72 millisec
7	5.04 millisec
8	40.32 millisec
9	0.36 sec
10	3.63 sec
11	39.92 sec
12	7.98 min
13	1.73 hr
14	24.22 hr
15	15.14 days
⋮	⋮
20	77,147 years

TABLE 15.2 Computer Times for Job Sequencing on a Computer 1,000 Times Faster

Number of Jobs	Computer Time
5	0.12 microsec
6	0.72 microsec
7	5.04 microsec
8	40.32 microsec
9	362.88 microsec
10	3.63 millisec
11	39.92 millisec
12	479.00 millisec
13	6.23 sec
14	87.18 sec
15	21.79 min
⋮	⋮
20	77.147 years

state of Texas—to a height of over *6,000 miles!* Now that's big. (Perhaps this is why mathematicians use the exclamation point to indicate the factorial function.)

Now let us relate these big numbers to computation times. Suppose we have a "slow" computer that can examine 1,000,000 sequences per second and we wish to build a scheduling system that has a response time of no longer than one minute. Assuming we must examine every possible sequence to find the optimum, how many jobs can we sequence optimally? Table 15.1 shows the computation times for various numbers of jobs and indicates that 11 jobs is the maximum we can sequence in less than one minute.

Now suppose we purchase a computer that runs 1,000 times faster than our old "slow" one (i.e., it can examine one billion sequences per second). Now how many jobs can be examined in less than one minute? From Table 15.2 we see that the maximum problem size we can solve only increases to 13 jobs (or 14 if we allow the maximum time to increase to one and one-half minutes). A 1,000 fold increase in computer speed only results in an 18 percent increase in size of the largest problem that can be solved in the specified time. The basic conclusion is that even big increases in computer speed do not dramatically increase our power to solve nonpolynomial problems.

For comparison, we now consider problems that do not grow exponentially. These are called **polynomial** problems because the time to solve them can be bounded a polynomial function of problem size (for example, n^2, n^3, etc., where n is a measure of problem size).

As a specific example, consider the job dispatching problem described in Section 15.2.3 and suppose we wish to dispatch jobs according to the SPT rule. This requires us to sort the jobs in front of the workstation according to process time.[1] There are well-known algorithms for sorting a list of elements whose computation time (i.e., number of steps) is proportional to $n \log n$, where n is the number of elements being

[1] Actually, in practice we would probably maintain the queue in sorted order, so we would not have to resort it each time a job arrived. This would make the problem even simpler than we indicate here.

TABLE 15.3 Computer Times for Job Sorting on the Slow Computer

Number of Jobs	Computer Time
10	3.6 sec
11	4.1 sec
12	4.7 sec
⋮	⋮
20	9 4 sec
30	16.1 sec
⋮	⋮
80	55.2 sec
85	59.5 sec
90	63.8 sec
⋮	⋮
100	72.6 sec
200	167.0 sec

TABLE 15.4 Computer Times for Job Sorting on a Computer 1,000 Times Faster

Number of Jobs	Computer Time
1,000	1.1 sec
2,000	2.4 sec
3,000	3.8 sec
⋮	⋮
10,000	14.5 sec
20,000	31.2 sec
30,000	48.7 sec
35,000	57.7 sec
36,000	59.5 sec
⋮	⋮
50,000	85.3 sec
100,000	181.4 sec
200,000	384.7 sec

sorted. This function is clearly bounded by n^2, a polynomial. Therefore, dispatching has polynomial complexity.

Suppose, just for the sake of comparison, that on the slow computer of the previous example it takes the same amount of time to sort 10 jobs as it does to examine 10! sequences (that is, 3.6 seconds). Table 15.3 reveals how the sorting times grow for lists of jobs longer than 10. Notice that we can sort 85 jobs and still remain below one minute (as compared to 11 jobs for the sequencing problem).

Even more interesting is what happens when we purchase the computer that works 1,000 times faster. Table 15.4 shows the computation times and reveals that we can go from sorting 85 jobs on the slow computer to sorting around 36,000 on the fast one. This represents an increase of over 400 percent, as compared to the 18 percent increase we observed for the sequencing problem. Evidently, we gain a lot from a faster computer for the "easy" (polynomial) sorting problem, but not much for the "hard" (exponential) sequencing problem.

Implications for Real Problems. Because most real-world scheduling problems fall into the NP-hard category and tend to be large (e.g., involving hundreds of jobs and tens of machines), the above results have important consequences for manufacturing practice. Quite literally, they mean that it is impossible to solve many realistically sized scheduling problems optimally.[2]

Fortunately, the practical consequences are not quite so severe. Just because we cannot find the *best* solution does not mean that we cannot find a *good* one. In some ways, the nonpolynomial nature of the problem may even help, since it implies that there may

[2] A computer with as many bits as there are protons in the universe, running at the speed of light, for the age of the universe, would not have enough time to solve some of these problems. Therefore the word *impossible* is *not* an exaggeration.

be many candidates for a good solution. Reconsider the 25-job sequencing problem. If "good" solutions were extremely rare to the point that only one in a trillion of the possible solutions was good, there would still be more than 15 trillion good solutions. We can apply an approximate algorithm, called a **heuristic,** that has polynomial performance to search for one of these solutions. There are many types of heuristics, including such interestingly named techniques as *beam search, tabu search, simulated annealing,* and *genetic algorithms.* We will describe one of these (tabu search) in greater detail when we discuss bottleneck scheduling.

15.2.5 Good News and Bad News

We can draw a number of insights from this review of scheduling research that are useful to the design of a practical scheduling system.

The Bad News. We begin with the negatives. First, unfortunately, most real-world problems violate the assumptions made in the classic scheduling theory literature in at least the following ways:

1. There are always more than two machines. Thus Johnson's minimizing makespan algorithm and its many variants are not directly useful.
2. Process times are not deterministic. In Part II we learned that randomness and variability contribute greatly to the congestion found in manufacturing systems. By ignoring this, scheduling theory may have overlooked something fundamental.
3. All jobs are *not* ready at the beginning of the problem. New jobs *do* arrive and continue arriving during the entire life of the plant. To pretend that this does not happen or to assume that we "clear out" the plant before starting new work is to deny a fundamental aspect of plant behavior.
4. Process times are frequently sequence-dependent. Often the number of setups performed depends on the sequence of the jobs. Jobs of like or similar parts can usually share a setup while dissimilar jobs cannot. This can be an important concern when scheduling the bottleneck process.

Second, real-world production scheduling problems are hard (in the NP-hard sense), which means

1. We cannot hope to find optimal solutions of many realistic-size scheduling problems.
2. Nonpolynomial approaches, like dispatching, may not work well.

The Good News. Fortunately, there are also positives, especially when we realize that much of the scheduling research suffers from type III error: solving the *wrong* problem. The formalized scheduling problems addressed in the operations research literature are models, not reality. The constraints assumed in these models are not necessarily fixed in the real world since, to some extent, we can control the problem by controlling the environment. This is precisely what the Japanese did when they made a hard scheduling problem much easier by reducing setup times. When we think along these lines, the failures as well as the successes of the scheduling research literature can lead us to useful insights, including the following.

Due dates: We do have some control over due dates; after all, someone in the company sets or negotiates them. We do not have to take them as given, although this is exactly what some companies and most scheduling problem formulations do. Section 15.3.2 presents a procedure for quoting due dates that are both achievable and competitive.

Job splitting: The SPT results for a single machine suggest that small jobs clear out more quickly than large jobs. Similarly, the mechanics of Johnson's algorithm call for a sequence that has a small job at both the beginning and the end. Thus, it appears that small jobs will generally improve performance with regard to average cycle time and machine utilization. However, in Part II we also saw that small batches result in lost capacity due to an increased number of setups. Thus, if we can somehow have large *process* batches (i.e., many units processed between setups) and small *move* batches (i.e., the number accumulated before moving to the next process), we can have both short cycle times and high throughput. This concept of lot splitting, which was illustrated in Chapter 9, thus serves to make the system less sensitive to scheduling errors.

Feasible schedules: An *optimal* schedule is really only meaningful in a mathematical model. In practice what we need is a *good, feasible* one. This makes the scheduling problem much easier because there are so many more candidates for a good schedule than for an optimal schedule. Indeed, as current research is beginning to show, various heuristic procedures can be quite effective in generating reasonable schedules.

Focus on bottlenecks: Because bottleneck resources can dominate the behavior of a manufacturing system, it is typically most critical to schedule these resources well. Scheduling the bottleneck(s) separately and then propagating the schedule to nonbottleneck resources can break up a complex large-scale scheduling problem into simpler pieces. Moreover, by focusing on the bottleneck we can apply some of the insights from the single-machine scheduling literature.

Capacity: As with due dates, we have some control over capacity. We can use some capacity controls (e.g., overtime) on the same time frame as that used to schedule production. Others (e.g., equipment or workforce changes) require longer time horizons. Depending on how overtime is used, it can simplify the scheduling procedure by providing more options for resolving infeasibilities. Also, if longer-term capacity decisions are made with an eye toward their scheduling implications, these, too, can make scheduling easier. Chapter 16 discusses aggregate planning tools that can help facilitate this.

With these insights in mind, we now examine some basic scheduling scenarios in greater detail. The methods we offer are not meant as ready-to-use solutions—the range of scheduling environments is too broad to permit such a thing—but rather as building blocks for constructing reasonable solutions to real problems.

15.2.6 Practical Finite-Capacity Scheduling

In this section we discuss some representative scheduling approaches, called variously **advanced planning systems** and **finite-capacity scheduling,** available in commercial software systems. Since the problems they address are large and NP-hard, all these make use of heuristics and hence none produces an optimal schedule (regardless of what the marketing materials might suggest). Moreover, these scheduling applications are generally additions to the MRP (material requirements planning) module within the ERP (enterprise resources planning) framework. As such, they attempt to take the planned order releases of MRP and schedule them through the shop so as to meet due dates, reduce the number of setups, increase utilization, decrease WIP, and so on. Unfortunately, if the planned order releases generated by MRP represent an infeasible plan, no amount of rescheduling can make it feasible. This is a major shortcoming of such "bolt-on" applications.

Finite-capacity scheduling systems typically fall into two categories: simulation-based and optimization-based. However, many of the optimization-based methods also make use of simulation.

Simulation-Based Scheduling. One way to avoid the NP-hard optimization problem is to simply ignore it. This can be done by developing a detailed and *deterministic* (i.e., no unpredictable variation in process times, no unscheduled outages, etc.) simulation model of the entire system. The model is then interfaced to the WIP tracking system of ERP to allow downloading of the current status of active jobs. Demand information is obtained from either the master production schedule module of ERP or another source. To generate a schedule, the model is run forward in time and records the arrival and departure of jobs at each station. Different schedules are generated by applying various **dispatching rules** at each station. These are evaluated according to selected performance measures to find the "best" schedule.

An advantage of the simulation approach is that it is easier to explain than most optimization-based methods. Since a simulator mimics the behavior of the actual system in an intuitive way, planners and operators alike can understand its logic. Another advantage is that it can quickly generate a variety of different schedules by simply changing dispatching rules and then reporting statistics such as machine utilization and the number of tardy jobs to the user. The user can choose from these the schedule that best fits his or her needs. For example, a custom job shop might be more interested in on-time delivery than in utilization, whereas a production system that uses extremely expensive equipment to make a commodity would be more interested in keeping utilization high.

However, there are also disadvantages. First, simulation requires an enormous amount of data that must be constantly maintained. Second, because the model does not account for variability, there can be large discrepancies between predicted and actual behavior. However, since virtually all finite-capacity scheduling procedures ignore variability, this problem is not limited to the simulation approach. The consequence is that to prevent error from piling up and completely invalidating the schedule over time it is important to regenerate the schedule frequently.

A third problem is that because there is no general understanding of when a given dispatching rule works well, finding an effective schedule is a trial-and-error process. Also, because dispatching rules are inherently myopic, it may be that no dispatching rule generates a good schedule.

Finally, the simulation approach, like the optimization approach, is generally used as an add-on to MRP. In a simulation-based scheduler, MRP release times are used to define the work that will be input into the model. However, if the MRP release schedule is inherently infeasible, simple dispatching cannot make it feasible. Something else—either capacity or demand—must change. But simulation-based scheduling methods are not well suited to suggesting ways to make an infeasible schedule feasible. For this an entirely different procedure is needed, as we discuss in Section 15.5.

Optimization-Based Scheduling. Unlike classical optimization, optimization-based scheduling techniques use heuristic procedures for which there are few guarantees of performance. The difference between optimization-based and simulation-based scheduling techniques is that the former uses some sort of algorithm to actively search for a good schedule.We will provide a short overview of these techniques and refer the reader interested in more details to a book devoted to the subject by Morton and Pentico (1993).

There are a variety of ways to simplify a complex scheduling problem to facilitate a tractable heuristic. One approach is to use a simulation model, like the simulation-based methods discussed, and have the system search for parameters (e.g., dispatching rules) that maximize a specified objective function. However, since it only searches over a partial set of policies (e.g., those represented by dispatching rules), it is not a true optimization approach.

An approach that makes truer use of optimization is to reduce a line or shop scheduling problem to a single-machine scheduling problem by focusing on the bottleneck. We refer to heuristics that do this as "OPT-like" methods, since the package called "Optimized Production Technique" developed in the early 1980s by Eliyahu Goldratt and others was the first to popularize this approach. Although OPT was sold as a "black box" without specific details on the solution approach, it involved four basic stages:

1. Determine the bottleneck for the shop.
2. Propagate the due date requirements from the end of the line back to the bottleneck using a fixed lead time with a time buffer.
3. Schedule the bottleneck most effectively.
4. Propagate material requirements from the bottleneck backward to the front of the line using a fixed lead time to determine a release schedule.

Simons and Simpson (1997) described this procedure in greater detail, extending it to cases in which there are multiple bottlenecks and when parts visit a bottleneck more than once. Because they use an objective function that weights due date performance and utilization, OPT-like methods can be used to generate different types of schedules by adjusting the weights.

An entirely different optimization-based heuristic is **beam search,** which is a derivative of the branch-and-bound technique mentioned earlier. However, instead of checking each branch generated, beam search checks only relatively few branches that are selected according to some sort of "intelligent" criteria. Consequently, it runs much faster than branch-and-bound but cannot guarantee an optimal solution.

An entire class of optimization-based heuristics are those classed as **local search techniques,** which start with a given schedule and then search in the "neighborhood" of this schedule to find a better one. It turns out that "greedy" techniques, which always select the best nearby schedule, do not work well. This is because there are many schedules that are not very good overall but are best in a very small local neighborhood. A simple greedy method will usually end up with one of these and then quit.

Several methods have been proposed to avoid this problem. One of these is called **tabu search** because it makes the most recent schedules "taboo" for consideration, thereby preventing the search from getting stuck with a locally good but globally poor schedule. Consequently, the search will move away from a locally good schedule and, for awhile, may even get worse while searching for a better schedule. Another method for preventing local optima is use of **genetic algorithms** that consider the characteristics of several "parent" schedules to generate new ones and then allow only good "offspring" to survive and "reproduce" new schedules. Still another is **simulated annealing,** which selects candidate schedules in a manner that loosely mimics the gradual cooling of a metal to minimize stress. In simulated annealing, wildly random changes to the schedule can take place early in the process, where some improve the schedule and others make it worse. However, as time goes on, the schedule becomes less volatile (i.e., is "cooled") and the approach becomes more and more greedy. Of course, all local search methods "remember" the best schedule that has been found at any point, in case no better schedule can be found. We will contrast one of these techniques (tabu search) with the greedy method in Section 15.4 on bottleneck sequencing.

Optimization-based heuristics can be applied in many different ways to a variety of scheduling problems. Within a factory, the most common problem formulations are (1) minimizing some measure of tardiness, (2) maximizing resource utilization, and (3) some combination of these. We have seen that tardiness problems are extremely difficult even for one machine. Utilization (e.g., makespan) problems are a little easier. But they also become intractable when there are more than two machines. So developing effective heuristics is not simple. Pinedo and Chao (1999) give details on which methods work well in various settings and how they can be implemented effectively.

One problem with optimization-based scheduling is that many practical scheduling problems are not really optimization problems at all but, rather, are better characterized as *satisficing* problems. Most scheduling professionals would not consider a schedule that has several late jobs as optimal. This is because some constraints, such as due dates and capacity, are not *hard* constraints but are more of a "wish list." Although the scheduler would rather not add capacity, it could be done if required to meet a set of demands. Likewise, it might be possible to split jobs or postpone due dates if required to obtain a feasible schedule. It is better to have a schedule that is implementable than one that optimizes an abstract objective function but cannot possibly be accomplished.

As with simulation-based scheduling, optimization-based scheduling has found useful implementation despite its drawbacks. A number of firms have been successful in combining such software (some developed in-house) with MRP II systems to assist planners. Arguello (1994) provides an excellent survey of finite-capacity scheduling software (both optimization-based and simulation-based) used in the semiconductor industry. Since most of this software has also been applied in other industries, the survey is relevant to non-semiconductor practitioners as well.

15.3 Linking Planning and Scheduling

Within an enterprise resources planning system, the MRP module generates planned order releases based on fixed lead times and other simplifying assumptions. As has been discussed before, this often results in an infeasible schedule. Also, because finite-capacity scheduling is far from a mature technology, many of the advanced planning systems found in modern ERP systems are complex and cumbersome. The time required to generate a capacity-feasible schedule makes it impractical to do so with any kind of regularity.

These problems have led to the practice of treating material planning (e.g., MRP), capacity planning (e.g., capacity requirements planning (CRP)), and production execution (e.g., order release and dispatching) separately in terms of time, software, and personnel. For example, material requirements planning determines what materials are needed and provides a rudimentary schedule without considering capacity. Then the capacity planning function performs a check to see if the needed capacity exists. If not, either the user (e.g., by iterating CRP) or the system (e.g., by using some advanced planning systems) attempts to reschedule the releases. But because capacity was not considered when material requirements were set, the capacity planning problem may have been made unnecessarily difficult (indeed, impossible). The problem is further aggravated by the common practice of having one department (e.g., production control) generate the production plan (both materials and capacity) which is then handed off to a different department (manufacturing) to execute.

An important antidote to the planning/execution disconnect is cycle time reduction. If cycle times are short (e.g., the result of variability reduction and/or use of some sort of pull system), the short-term production planning function (i.e., committing to demands) can provide the production schedule.[3] However, before that can be done, the production planning and scheduling problem must be recast from one of *optimization*, subject to given constraints of capacity and demand, to one of *feasibility analysis*, to determine what must be done in order to have a practical production plan. This requires a procedure that analyzes both material and capacity requirements simultaneously. This can be done in theory with a large mathematical programming model. However, such formulations are usually slow and therefore prohibit making frequent feasibility checks as the situation evolves. We present a practical heuristic method that provides a quick feasibilty check in Section 15.5.2.

The remainder of this chapter focuses on issues central to the development of practical scheduling procedures. In the remainder of this section we consider techniques for making scheduling problems easier, namely, effective batching and due date quoting. Section 15.4 deals with bottleneck scheduling in the context of CONWIP lines. For more general situations, we provide a method that considers material and capacity simultaneously in Section 15.5. Finally, in Section 15.6 we show how to use scheduling (which is inherently "push" in nature) within a pull environment.

15.3.1 Optimal Batching

In Chapter 9 we observed that process batch sizes can have a tremendous impact on cycle time. Hence, batching can also have a major influence on scheduling. By choosing batch sizes wisely, to keep cycle times short, we can make it easier for a schedule to meet due dates. We now develop methods for determining batch sizes that minimize cycle time.

Optimal Serial Batches. Figure 15.1 shows the relation between average cycle time and the serial batch size. With the formulas developed in Chapter 9, we could plot the total cycle time and find an optimal batch size for a single part at a station. However, this would be cumbersome and is of little value when we have multiple parts that interact with one another. So instead we derive a simple procedure that first finds the (approximately) optimal utilization of the station and then uses this to compute the serial batch size. We do this first for the case of a single part and then extend the approach to multiproduct systems.

Technical Note: Optimal Serial Process Batch Sizes

We first consider the case in which the product families are identical with respect to process and setup times and arrivals are Poisson. The problem is to find the serial batch size that minimizes total cycle time at a single station. This batch size should be good for the line if only one station has significant setups and tends to be the bottleneck.

Using the notation from Chapter 9, the effective process time for a batch is $t_e = s + kt$, and utilization is given by

$$u = \frac{r_a}{k}(s + kt)$$

[3] Long-term production planning, also known as aggregate planning, is used to set capacity levels, plan for workforce changes, etc. (see Chapter 16).

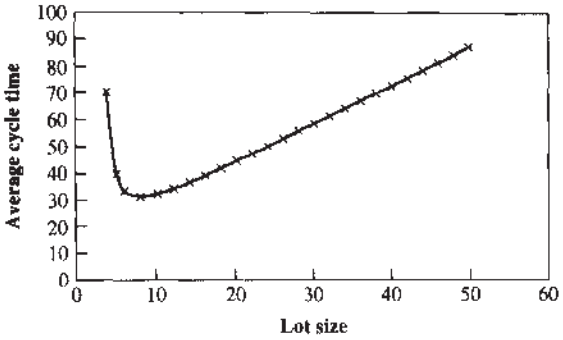


Figure 15.1

Average cycle time versus serial batch size

Now define the "utilization without setups" as $u_0 = r_a t$. A little algebra shows that the effective process time of a batch can be written

$$t_e = \frac{su}{u - u_0}$$

Since we are assuming Poisson arrivals (a good assumption if products arrive from a variety of sources), the arrival squared coefficient of variation (SCV) is $c_a^2 = 1$ and average cycle time is

$$\text{CT} = \left(\frac{1 + c_e^2}{2}\right)\left(\frac{u}{1 - u}\right)\frac{su}{u - u_0} + \frac{su}{u - u_0} \tag{15.1}$$

Written in this way, cycle time is a function of u only, instead of k and u. So minimizing cycle time boils down to finding the optimal station utilization. We do this by taking the derivative of (15.1) with respect to u, setting it equal to zero, and solving, which yields,

$$u^* = \frac{\alpha u_0 + \sqrt{\alpha^2 u_0^2 + [\alpha(1 + u_0) + 1]u_0}}{\alpha(1 + u_0) + 1} \tag{15.2}$$

where $\alpha = (1 + c_e^2)/2 - 1$. Note that in the special case where $c_e^2 = 1$ we have that $\alpha = 0$ and

$$u^* = \sqrt{u_0} \tag{15.3}$$

But even when c_e^2 is not equal to one, the value of u^* generally remains close to $\sqrt{u_0}$. For example, when $u_0 = 0.5$ and $c_e^2 = 15$, the difference is less than five percent. Moreover, the closer u_0 is to one (i.e., the higher the utilization of the system without setups), the smaller the difference between u^* and $\sqrt{u_0}$ for all c_e^2 (see Spearman and Kröckel 1999).

To obtain the batch size, recall that

$$u^* = \frac{r_a}{k^*}(s + k^* t) = \frac{r_a s}{k^*} + u_0$$

and solve for k^*.

The above analysis shows that a good approximation of the serial batch size that minimizes cycle time at a station is

$$k^* = \frac{r_a s}{u^* - u_0} \approx \frac{r_a s}{\sqrt{u_0} - u_0} \tag{15.4}$$

where $u_0 = r_a t$. We illustrate this with the following example.

Example: Optimal Serial Batching (Single Product)
Consider the serial batching example in Section 9.4 and shown in Figure 15.1. The utilization without considering setups u_0 is

$$u_0 = r_a t = (0.4 \text{ part/hour})(1 \text{ hour}) = 0.4$$

So, by Equation (15.3), optimal utilization is approximately

$$u^* = \sqrt{u_0} = \sqrt{0.4} = 0.6325$$

and by Equation (15.4) the optimal batch size is

$$k^* = \frac{r_a s}{u^* - u_0} = \frac{0.4(5)}{0.6325 - 0.4} = 8.6 \approx 9$$

From Figure 15.1, we see that this is indeed very close to the true optimum of eight. The difference in cycle time is less than one percent.

The insight that the optimal station utilization is very near to the square root of the utilization without setups is extremely robust. This allows it to be used as the basis for a serial batch-setting procedure in more general multiple-product family systems. We develop such an approach in the next technical note.

Technical Note—Optimal Serial Batches with Multiple Products
To model the multiproduct case we define the following:

n = number of products
i = index for products, $i = 1, \ldots, n$
r_{ai} = demand rate for product i (parts per hour)
t_i = mean time to process one part of product i (hours)
c_{ti}^2 = SCV of time to process one part of product i
s_i = mean time to perform setup when changing to product i (hours)
c_{si}^2 = SCV of time to perform setup when changing to product i
t_e = effective process time averaged over all products (hours)
c_e^2 = SCV of effective process time averaged over all products
u_0 = $\sum_i r_{ai} t_i$ = station utilization without setups
u = station utilization
k_i = lot size for product i

We can use the *VUT* equation to compute cycle time at the station as

$$\text{CT} = \left(\frac{Vu}{1-u} + 1\right) t_e \tag{15.5}$$

where $V = (1 + c_e^2)/2$. To use this, we must compute u, t_e and c_e from the individual part data. Utilization is given by

$$u = \sum_{i=1}^{n} \frac{r_{ai}}{k_i}(s_i + k_i t_i)$$

The effective process time is, in a sense, the "mean of the means." In other words, if the mean process time for a batch of i is $s_i + k_i t_i$ and the probability that the batch is for product i is π_i, then the effective process time is

$$t_e = \sum_{i=1}^{n} \pi_i (s_i + k_i t_i) \tag{15.6}$$

The probability that the batch is of a given product type is the ratio of that type's arrival rate to the total arrival rate

$$\pi_i = \frac{r_{ai}/k_i}{\sum_{j=1}^{n}(d_j/k_j)} \tag{15.7}$$

Using standard stochastic analysis, we compute the variance of the effective run time σ_e^2 as

$$\sigma_e^2 = \sum_{i=1}^{n} \pi_i(k_i c_{ti}^2 t_i^2 + c_{si}^2 s_i^2) + \left[\sum_{i=1}^{n} \pi_i (s_i + k_i t_i)^2 - t_e^2\right] \tag{15.8}$$

and hence the effective SCV is $c_e^2 = \sigma_e^2/t_e^2$.

Now, assuming as we did in the single-product case that $u^* = \sqrt{u_0}$ is a good approximation of the optimal utilization, the lot-sizing problem reduces to finding a set of k_i values that achieve u^* and keep c_e^2 and t_e small. From Equation (15.5) it is clear that this will lead to a small cycle time. Note that if all the values of $s_i + k_i t_i$, that is, all the average run lengths, were equal, the term in square brackets in Equation (15.8) would be zero. Thus, one way to keep both t_e and c_e^2 small is to minimize the average run length and to make all the run lengths the same. We can express this as the following optimization problem.

$$\begin{aligned} &\text{Minimize} && L \\ &\text{Subject to:} && s_i + k_i t_i \le L \qquad \text{for } i = 1, \ldots, n \\ &&& \sum_{i=1}^{n} \frac{r_{ai}}{k_i}(s_i + k_i t_i) = u^* \end{aligned}$$

The solution can be obtained from

$$\begin{aligned} s_i + k_i t_i &= L \\ k_i &= \frac{L - s_i}{t_i} \end{aligned} \tag{15.9}$$

Then solve for L, using the constraint

$$\begin{aligned} \sum_{i=1}^{n} \frac{r_{ai}}{k_i}(s_i + k_i t_i) &= u^* \\ \sum_{i=1}^{n} \frac{r_{ai} s_i}{k_i} &= u^* - u_0 \\ \sum_{i=1}^{n} \frac{r_{ai} s_i t_i}{L - s_i} &= u^* - u_0 \end{aligned}$$

If the setup times are all close to the mean setup time, which we denote by $\bar{s}$, then we can solve for L as follows.

$$L = \frac{\sum_{i=1}^{n} r_{ai} s_i t_i}{u^* - u_0} + \bar{s} \tag{15.10}$$

Substituting this into Equation (15.9) yields approximately optimal batch sizes.

The above analysis shows that the serial batch size for product i that minimizes cycle time at a station with multiple products and setups is

$$k_i^* = \frac{L - s_i}{t_i} \tag{15.11}$$

where L is computed from Equation (15.10).

Example: Optimal Serial Batching (Multiple Products)

Consider an industrial process in which a blender blends three different products. Demand for each product is 15 blends per month and is controlled by an MRP system that uses a constant batch size for each product. Whenever the blender is switched from one product to another, a cleanup is required. Products A and B take four hours per blend and eight hours for cleanup. Product C requires eight hours per blend and 12 hours for cleanup. All process and setup times have a coefficient of variation of one-half. The blender is run two shifts per day, five days per week. With one hour lost for each shift and 52/12 weeks per month, this averages out to 303.33 hours per month.

In keeping with conventional wisdom (e.g., the EOQ model) that longer changeovers should have larger batch sizes, the firm is currently using batch sizes of 20 blends for products A and B and 30 blends for product C. The average cycle time through the process is currently around 32 shop days. But could they do better?

Converting demand to units of hours yields $r_{ai} = 15/303.33 = 0.0495$ blend per hour for all three products. The utilization without setups is therefore

$$u_0 = 0.0495(4 + 4 + 8) = 0.7912$$

Hence, the optimal utilization is $u^* = \sqrt{u_0} = \sqrt{0.7912} = 0.8895$.

The average setup time is $\bar{s} = (8 + 8 + 12)/3 = 9.33$ hours, so the sum needed in Equation (15.10) is

$$\sum_{i=1}^{3} r_{ai} s_i t_i = 0.0495[8(4) + 8(4) + 12(8)] = 7.912$$

and hence

$$L = \frac{7.912}{0.8895 - 0.7912} + 9.33 = 89.82$$

With this we can compute the approximately optimal batch sizes as follows.

$$k_A = k_B = \frac{L - s_A}{t_A} = \frac{89.82 - 8}{4} = 20.46 \approx 20$$

$$k_C = \frac{L - s_C}{t_C} = \frac{89.82 - 12}{8} = 9.73 \approx 10$$

Using these batch sizes results in an average cycle time of 20.28 days, a decrease of over 36 percent. Doing a complete search over all possible batch sizes shows that this is indeed the optimal solution.

Note that the batch size for part C is *smaller* than that for A and B. EOQ logic, which was developed assuming separable products, suggests that C should have a larger batch size because it has a longer setup time. But to keep the run lengths equal across products, we need to reduce the batch size of C.

Optimal Parallel Batches. A machine with parallel batching is a true batch machine, such as a heat treat oven in a machine shop or a copper plater in a circuit-board plant. In these cases, the process time is the same regardless of how many parts are processed at once (the batch size).

In parallel batching situations, the basic tradeoff is between effective capacity utilization, for which we want large batches, and minimal wait-to-batch time, for which we want small batches. If the machine is a bottleneck, it is often best to use the largest batch possible (size of the batch operation). In nonbottlenecks, it can be best (in terms of cycle time) to process a partial batch. The following technical note describes a procedure for determining the optimal parallel batch size at a single station.

Technical Note—Optimal Parallel Batches

To find a batch size that minimizes cycle time at a parallel batch operation, it is convenient to find the best utilization and then translate this to a batch size, as we did in the case of serial process batching.

To do this, we make use of the following notation:

r_a = arrival rate (parts per hour)
c_a = coefficient of variation (CV) of interarrival times
t = time to process a batch (hours)
c_e = effective CV for processing time of a batch
B = maximum batch size (number of parts that can fit into process)
$u_m = r_a t$ = utilization resulting from batch size of one
u = station utilization
k = parallel batch size

Note that utilization is given by $u = r_a/(k/t)$, which must be less than one for the station to be stable. We can use $u_m = r_a t$ to rewrite this as $u = u_m/k$, which implies the batch size is $k = u_m/u$.

Recall, from Chapter 9, that the total time spent in a parallel batch operation includes wait-to-batch time (WTBT), queue time, and the time of the operation itself, which can be written

$$\begin{aligned}\mathrm{CT} &= \mathrm{WTBT} + \mathrm{CT}_q + t \\ &= \frac{k-1}{2r_a} + \left(\frac{c_a^2/k + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t + t \\ &= \frac{k-1}{2ku}t + \left(\frac{c_a^2/k + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t + t \qquad (15.12)\end{aligned}$$

where the last equality follows from the fact that $r_a = uk/t$.

Substitution of $k = u_m/u$ allows us to rewrite Equation (15.12) as

$$\mathrm{CT} = \left(\frac{u_m/u - 1}{2u_m} + \frac{c_a^2 u/u_m + c_e^2}{2}\frac{u}{1-u} + 1\right)t \qquad (15.13)$$

Unfortunately, minimizing CT with respect to utilization does not yield a simple expression. So to approximate, we will let $\beta = c_a^2 u/u_m$ and assume that this term can be treated as a constant. Our justification for this is that when k is large, u/u_m will be small, which will make β negligible. This reduces the expression for cycle time to

$$\begin{aligned}\mathrm{CT} &\approx \left(\frac{1}{2u} - \frac{1}{2u_m} + \frac{\beta + c_e^2}{2}\frac{u}{1-u} + 1\right)t \\ &= \left(\frac{y(u)}{2} - \frac{1}{2u_m} + 1\right)t \qquad (15.14)\end{aligned}$$

where

$$y(u) = \frac{1}{u} + \frac{\beta + c_e^2 u}{1-u}$$

Minimizing Equation (15.14) is equivalent to minimizing $y(u)$ with respect to u, which is fairly easy. Taking the derivative of $y(u)$ with respect to u, setting it equal to zero, and solving yields

$$u^* = \frac{1}{1 + \sqrt{\beta + c_e^2}} \tag{15.15}$$

If, as we suggested it might be, β is close to zero, then the optimal utilization reduces to

$$u^* \approx \frac{1}{1 + c_e} \tag{15.16}$$

When c_e^2 is not too small, dropping the $\beta = c_a^2 u/u_m$ term does not have a large impact and equation (15.16) is a fairly good approximation. However, when c_e^2 is small, dropping this term significantly changes the problem. Indeed, when $c_e^2 = 0$, equation (15.16) suggests that the optimal utilization is equal to one! Of course, we know that this is not reasonable, since if there is any variability in the arrival process, the queue will blow up.

So, to go back and reintroduce the β term, we substitute the approximate expression for u^* from equation (15.16) into $c_a^2 u/u_m$, so that

$$\beta = \frac{c_a^2}{u_m(1 + c_e)}$$

and

$$u^* = \frac{1}{1 + \sqrt{c_a^2/[u_m(1 + c_e)] + c_e^2}} \tag{15.17}$$

Once we have the optimal utilization u^*, we can easily find the optimal batch size k^* from $k = u_m/u$.

Thus, we have that the process batch size that minimizes cycle time at a parallel batch station is

$$k^* = \frac{u_m}{u^*} \tag{15.18}$$

where $u_m = r_a t$ and u^* is computed using Equation (15.17). To obtain an integer batch size, we will use the convention of rounding *up* the value from Equation (15.18). This will tend to offset some of the error introduced by the approximations made in the technical note.

In addition to a computational tool, Equations (15.17) and (15.16) yield some qualitative insight. They indicate that the more variability we have at the station, the less utilization it can handle. Specifically, as c_e or c_a increases, the optimal utilization of the system decreases. This is a consequence of the factory physics results on variability and utilization, which showed that these two factors combine to degrade performance. Hence, when we are optimizing performance, we must offset more variability via less utilization.

We illustrate the use of the formula for parallel batch sizing in the following example.

Example: Optimal Parallel Batching

Reconsider the burn-in operation discussed in Section 9.4, in which a facility tests medical diagnostic units in an operation that turns the units on and runs them in a temperature-controlled room for 24 hours regardless of how many units are being burned in. The burn-in room can hold 100 units at a time, and units arrive to burn in at a rate of one per hour (24 per day). Figure 9.6 plots cycle time versus batch size for this example and shows that cycle time is minimized at a batch size of 32, which achieves a cycle time of 42.88 hours.

Now consider the situation using the above optimal batch-sizing formulas. The arrival rate is $r_a = 1$ per hour and arrivals are Poisson, so $c_a = 1$. The process time is

$t = 24$ hours, and it has variability such that $c_e = 0.25$. So for stability we require a batch size $k > u_m = r_a t = 24$, which implies that the minimum batch size is 25.

However, if we use a batch size of 25, we get

$$u = \frac{r_a}{k/t} = \frac{1}{25/24} = 0.96$$

$$\text{WTBT} = \frac{k-1}{2r_a} = \frac{25-1}{2(1)} = 12 \text{ hours}$$

$$\text{CT}_q = \left(\frac{c_a^2/k + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t$$

$$= \left(\frac{1/25 + 0.25^2}{2}\right)\left(\frac{0.96}{1-0.96}\right)24 = 29.52 \text{ hours}$$

Hence, the average cycle time through the heat treat operation will be

$$\text{CT} = \text{WTBT} + \text{CT}_q + t = 12 + 29.52 + 24 = 65.52 \text{ hours}$$

Now consider the other extreme and let $k = 100$, the size of the burn-in room.

$$u = \frac{r_a}{k/t} = \frac{1}{100/24} = 0.24$$

$$\text{WTBT} = \frac{k-1}{2r_a} = \frac{100-1}{2(1)} = 49.5 \text{ hours}$$

$$\text{CT}_q = \left(\frac{c_a^2/k + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t$$

$$= \left(\frac{1/100 + 0.25^2}{2}\right)\left(\frac{0.24}{1-0.24}\right)24 = 0.27 \text{ hour}$$

So the average cycle time through the heat treat operation will be

$$\text{CT} = \text{WTBT} + \text{CT}_q + t = 49.5 + 0.27 + 24 = 73.77 \text{ hours}$$

Now to find the optimal batch size, we first compute the optimal utilization.

$$u^* = \frac{1}{1 + \sqrt{c_a^2/[u_m(1+c_e)] + c_e^2}}$$

$$= \frac{1}{1 + \sqrt{1/[24(1+0.25)] + 0.25^2}}$$

Then we use Equation (15.18) to compute

$$k^* = \frac{u_m}{u^*} = \frac{24}{0.7636} = 31.43 \approx 32$$

Note that this is exactly the optimal batch size we observed in Figure 9.6. Furthermore, the minimum batch size yields a cycle time that is 53 percent higher than the optimum, while the maximum batch size yields one that is 72 percent greater than optimal. Clearly, batching can have a significant impact on cycle times in *parallel batch operations*.

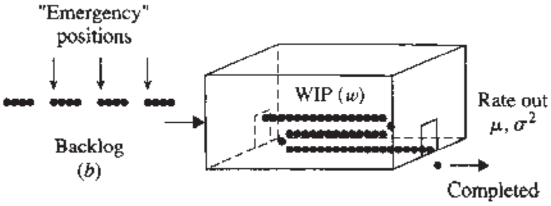


FIGURE 15.2

Schematic of method for quoting lead times

15.3.2 Due Date Quoting

Variability reduction (Chapter 9), pull production (Chapter 10), and efficient lot-sizing methods (previously described) all make a production system easier to schedule. Another technique for simplifying scheduling is due date quoting. Since scheduling problems that involve due dates are extremely hard, while the due date–setting problem can be relatively easy, this would seem worthwhile. Of course, in the real world, implementation is more than a matter of mathematics. Developing a due date–quoting system may involve a much more difficult problem—getting manufacturing and salespeople to talk to one another.

In addition to personnel issues, the difficulty of the due date–quoting problem depends on the manufacturing environment. To be able to specify reasonable due dates, we must be able to predict when jobs will be completed given a specified schedule of releases. If the environment is so complex that this is difficult, then due date quoting will also be difficult. However, if we simplify the environment in a way that makes it more predictable, then due date quoting can be made straightforward.

Quoting Due Dates for a CONWIP Line. One of the most predictable manufacturing systems is the CONWIP line. As we noted previously, CONWIP behavior can be characterized via the conveyor model. This enables us to develop a simple procedure for quoting due dates.

Consider a CONWIP line that maintains w standard units[4] of WIP and whose output in each period (e.g., shift, day) is steady with mean μ and variance σ^2. Suppose a customer places an order that represents c standard units of work, and we are free to specify a due date. To balance responsiveness with dependability, we want to quote the earliest due date that ensures a service level (probability of on-time delivery) of s. Of course, the due date that will achieve this depends on how much work is ahead of the new order. This in turn depends on how customer orders are sequenced. One possibility is that jobs are processed in first-come, first-serve order, in which case we let b represent the current backlog (i.e., number of standard jobs that have been accepted but not yet released to the line). Alternatively, "emergency slots" for high-priority jobs could be maintained (see Figure 15.2) by quoting due dates for some lower-priority jobs as if there were "placeholder" jobs already ahead of them. In this case, we define b to represent the units of work until the first emergency slot.

In either case, the customer order will be filled after $m = w + b + c$ standard units of work are output by the line. Hence the problem of finding the earliest due date that guarantees a service level of s is equivalent to finding the time within which we are s percent certain of being able to complete m standard units of work. We derive an expression for this time in the following technical note.

[4] A standard unit of WIP is one that requires a certain amount of time at the bottleneck of the line. Thus, CONWIP maintains a constant workload in the line, as measured by time on the bottleneck.

Technical Note—Due Date Quoting for a CONWIP Line

Let X_t be a random variable representing the amount of work (in standard units) completed in period t. Assume that X_t, $t = 1, 2, \ldots$, are independent and normally distributed with mean μ and variance σ^2. To guarantee completion by time ℓ with probability s, the following must be true:

$$P\left\{\sum_{t=1}^{\ell} X_t \leq m\right\} = 1 - s$$

Note that since the means and variances of independent random variables are additive, the amount of work completed by time ℓ is given by

$$\sum_{t=1}^{\ell} X_t \sim \mathrm{N}(\ell\mu,\ \ell\sigma^2)$$

That is, it is normally distributed with mean $\ell\mu$ and variance $\ell\sigma^2$. Hence,

$$P\left\{Z \leq \frac{m - \ell\mu}{\sqrt{\ell}\sigma}\right\} = 1 - s$$

where Z is the standard 0–1 normal random variable.

Therefore,

$$\frac{m - \ell\mu}{\sqrt{\ell}\sigma} = z_{1-s} \tag{15.19}$$

where z_{1-s} is obtained from a standard normal table.

We can rewrite Equation (15.19) as

$$\ell^2\mu^2 - (2\mu m + z_{1-s}^2\sigma^2)\ell + m^2 = 0 \tag{15.20}$$

which can be solved by using the quadratic equation. There are two roots to this equation; as long as $s \geq 0.5$, the larger one should always be used. This yields Equation (15.21).

The minimum quoted lead time for a new job consisting of c standard units that is sequenced behind a backlog of b standard units in a CONWIP line with a WIP level of w necessary to guarantee a service level of s is given by

$$\ell = \frac{m}{\mu} + \frac{z_{1-s}^2\sigma^2\left[1 + \sqrt{4\mu m/(z_{1-s}^2\sigma^2) + 1}\right]}{2\mu^2} \tag{15.21}$$

where $m = w + b + c$.

A possible criticism of the above method is that it is premised on service. Hence, a job that is one day late is considered just as bad as one that is one year late. A measure that better tracks performance from a customer perspective is tardiness. Fortunately, it turns out that quoting each job with the same service level also yields the minimum expected quoted lead time subject to a constraint on average tardiness (see Spearman and Zhang 1999).

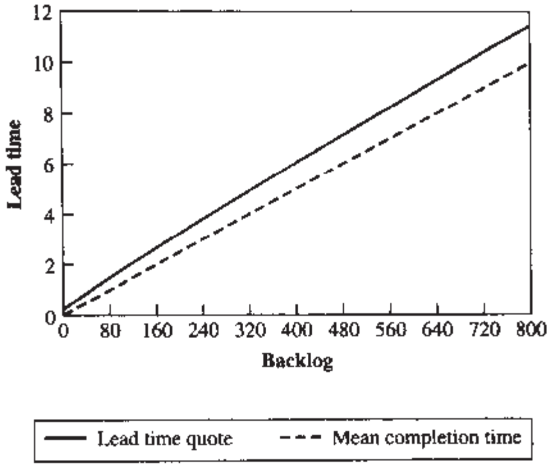
Furthermore, to simplify implementation with little loss in performance, Equation (15.21) can be replaced by

$$\ell = \frac{m}{\mu} + \text{planned inventory time} \tag{15.22}$$

where planned inventory time can be adjusted by trial and error to achieve acceptable service (see Hopp and Roof 1998).

Figure 15.3

Quoted lead times versus the backlog



Example: Due Date Quoting

Suppose we have a CONWIP line that maintains 320 standard units of WIP and has an average output of 80 units per day with a standard deviation of 15 units. The line receives a high-priority order representing 20 standard units, and the first available emergency slot on the backlog is 100 jobs from the start of the line. We want to quote a due date with a service level of 99 percent.

To use Equation (15.21), we observe that $\mu = 80$, $\sigma^2 = 225$ (or, 15^2), $w = 320$, $b = 100$, and $c = 20$, so that $m = 440$. The value for $z_{1-s} = z_{0.01} = -2.33$ is found in a standard normal table. Thus,

$$\begin{aligned}\ell &= \frac{m}{\mu} + \frac{z_s^2\sigma^2\left[1 + \sqrt{4\mu m/(z_s^2\sigma^2) + 1}\right]}{2\mu^2}\\ &= \frac{440}{80} + \frac{(-2.33^2)(225)\left\{1 + \sqrt{4(80)(440)/[(-2.33)^2(225)] + 1}\right\}}{2(80^2)}\\ &= 6.62\end{aligned}$$

and so we quote seven days to the customer.

Notice that the mean time to complete the order is $m/\mu = 440/80 = 5.5$ days. The additional one and one-half days represent **safety lead time** used as a buffer against the variability in the production process.

Figure 15.3 shows the lead time quotes as a function of total backlog m. The dashed line shows the mean completion time m/μ, which is what would be quoted if there were no variance in the production rate. The difference between the solid and dotted lines is the safety lead time, which we note increases in the backlog level. The reason is that the more work that must be completed to fill an order, the greater the variability in the completion time, and hence the higher the required safety lead time.

In an environment with multiple CONWIP routings, a similar set of computations would be performed for each routing in the plant. The only data needed are the first two moments of the production rate for the routing, the current WIP level (a constant under CONWIP), and the current status of the backlog. These data should be maintained in a central location accessible to both sales and manufacturing. Sales needs the information to quote due dates; manufacturing needs it to determine what to start next. Manufacturing can also track production against a backlog established by sales (e.g., the statistical throughput control procedure described in Chapter 14). The overall result will be due dates that are competitive, achievable, and consistent with manufacturing parameters.

15.4 Bottleneck Scheduling

A main conclusion of the scheduling research literature is that scheduling problems, particularly realistically sized ones, are very difficult. So it is common to simplify the problem by breaking it down into smaller pieces. One way to do this is by scheduling the bottleneck process by itself and then propagating that schedule to nonbottleneck stations. This is particularly effective in simple flow lines. However, bottleneck scheduling can also be an important component in more complex scheduling situations as well.

A major reason why restricting attention to the bottleneck can simplify the scheduling problem is that it reduces a multimachine problem to a single-machine problem. Recall from our discussion of scheduling research that simple sequences, as opposed to detailed schedules, are often sufficient for single-machine problems. Since a *schedule* presents information about when each job is to be run on each machine while a *sequence* only presents the order of processing the jobs, it is easier to compute a sequence. Furthermore, because schedules become increasingly inaccurate with time, sequences can be more robust in practice.

The scheduling problem can be further simplified if the manufacturing environment is made up of CONWIP lines. As we know (Chapter 13), a CONWIP line can be characterized as a conveyor with rate r_b^P (the practical production rate) and transit time T_0^P (minimum practical lead time). Since the parameters r_0^P and T_0^P are adjusted to include variability effects such as failures, variable process times, and setups, and because safety capacity (overtime) is used to ensure that the line achieves its target rate each period (day, week, or whatever), the deterministic conveyor model is a good approximation of the stochastic production system. Thus, by focusing on the bottleneck in a CONWIP line, we effectively reduce a very hard multistation stochastic scheduling problem to a much easier single-station deterministic scheduling problem. Also, since we use first-in-system first-out (FISFO) dispatching at each station, it is a trivial matter to propagate the bottleneck sequence to the other stations—simply use the same sequence at all stations. This sequence is the **CONWIP backlog** to which we have referred in previous chapters. In this section, we discuss how to generate this backlog.

15.4.1 CONWIP Lines Without Setups

We begin by considering the simplest case of CONWIP lines—those in which setups do not play a role in scheduling. This could be because there are no significant setups between any part changes. Alternatively, it could be because setups are done periodically (e.g., for cleaning or maintenance) but do not depend on the work sequence. Sequencing a single CONWIP line without setups is just like scheduling the single machine with due dates that we discussed earlier and hence can be done with the earliest due date (EDD) rule. Results from scheduling theory show that the EDD sequence will finish all the jobs on time if it is possible to do so. Of course, what this really means is that jobs will finish on time in the *planned* schedule. We cannot know in advance if this will occur, since it depends on random events. But starting with a feasible plan gives us a much better chance at good performance in practice than does starting with an infeasible plan.

A slightly more complex situation is one in which two or more CONWIP lines share one or more workstations. Figure 15.4 shows such a situation in which (1) two CONWIP lines share a machine that also happens to be the bottleneck and (2) the lines produce

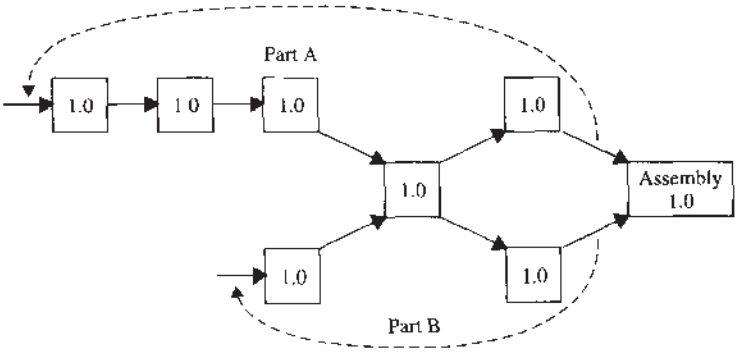


FIGURE 15.4
Two CONWIP lines sharing a common process center

components for an assembly operation. We consider this case because it starkly illustrates the issues involved. However, the scheduling is fundamentally the same as scheduling a system with the lines feeding separate finished goods inventory (FGI) buffers instead of assembly.

In both cases, we should sequence releases into the individual lines according to the EDD rule and use this sequence at all nonshared stations, just as we did for the separate CONWIP line case. This leaves the question of what sequence to use at the shared stations.

One might intuitively think that using first-in-first-out (FIFO) would work well. However, if there is variability in the process times, then, for example, eventually a string of A jobs will arrive at the shared resource before the matching B jobs. Using FIFO will therefore only create a queue of unmatched parts at the assembly operation. In extreme cases, this could actually cause the bottleneck to starve for work since so much WIP is tied up at assembly.

A better alternative is first-in-system-first-out (FISFO) dispatching at the shared resource. Under this rule, jobs are sequenced according to when they entered the system (i.e., the times their CONWIP cards authorized their release). Since the CONWIP cards authorize releases for matching parts (i.e., one A and one B) at assembly at the same time, this rule serves to sequence the shared machine according to the assembly sequence. Hence it serves to synchronize arrivals to assembly as closely as possible. Of course, when there are no B jobs to work on at the shared machine (due to an unusually long process time upstream, perhaps) it will process only A jobs. But as soon as it receives B jobs to work on, it will.

15.4.2 Single CONWIP Lines with Setups

The situation becomes more difficult when we consider a CONWIP line with setups at the bottleneck. Indeed, even determining whether a sequence exists that will satisfy all the due dates is to answer an NP-complete question.

To illustrate the difficulty of this problem and to suggest a solution approach, we consider the set of 16 jobs shown in Table 15.5. Each job takes one hour to complete, not including a setup. Setups take four hours and occur if we go from any job family to any other. The jobs in Table 15.5 are arranged in earliest due date order. As we see, EDD does not appear very effective here, since it results in 10 setups and 12 tardy jobs for an average tardiness of 10.4. To find a better solution, we clearly do not want to evaluate every possibility, since there are $16! = 2 \times 10^{13}$ possible sequences. Instead we seek a heuristic that gives a good solution.

Table 15.5 EDD Sequence

Job Number	Family	Due Date	Completion Time	Lateness
1	1	5	5	0
2	1	6	6	0
3	1	10	7	−3
4	2	13	12	−1
5	1	15	17	2
6	2	15	22	7
7	1	22	27	5
8	2	22	32	10
9	1	23	37	14
10	3	29	42	13
11	2	30	47	17
12	2	31	48	17
13	3	32	53	21
14	3	32	54	22
15	3	33	55	22
16	3	40	56	16

One possible approach is known as a **greedy algorithm.** Each step of a greedy algorithm considers all simple alternatives (i.e., pairwise interchanges of jobs in the sequence) and selects the one that improves the schedule the most. This is why it is called greedy. The number of possible interchanges (120 in this case) is much smaller than the total number of sequences, and hence this algorithm will find a solution quickly. The question of course is, How good will the solution be? We consider this below.

Checking the total tardiness for every possible exchange between two jobs in the sequence reveals that the biggest decrease is achieved by putting job 4 after job 5. As shown in Table 15.6, this eliminates two setups (going from family 1 to family 2 and back again). The average tardiness is now 5.0 with eight setups.

We repeat the procedure in the second step of the algorithm. This time, the biggest reduction in total tardiness results from moving job 7 after job 8. Again, this eliminates two setups by grouping like families together. The average tardiness falls to 1.2 with six setups. The third step moves job 10 after job 12, which eliminates one setup and reduces the average tardiness to one-half. The resulting sequence is shown in Table 15.7.

At this point, no further single exchanges can reduce total tardiness. Thus the greedy algorithm terminates with a sequence that produces three tardy jobs. The question now is, Could we have done better?

The answer, as shown in Table 15.8, which gives a feasible sequence, is yes. But must we evaluate all 16! possible sequences to find it? Mathematically speaking, we must. However, practically speaking, we can often find a better (even feasible) sequence by using a slightly more clever approach than the simple greedy algorithm.

To develop such a procedure, we observe that the problem with greedy algorithms is that they can quickly converge to a **local optimum**—a solution that is better than any other adjacent solutions, but not as good as a nonadjacent solution. Since the greedy algorithm considered only adjacent moves (pairwise interchanges), it is vulnerable to getting stuck at a local optimum. This is particularly likely because *NP-hard problems*

TABLE 15.6 Sequence after First Swap in Greedy Algorithm

Job Number	Family	Due Date	Completion Time	Lateness
1	1	5	5	0
2	1	6	6	0
3	1	10	7	−3
5	1	15	8	−7
4	2	13	13	0
6	2	15	14	−1
7	1	22	19	−3
8	2	22	24	2
9	1	23	29	6
10	3	29	34	5
11	2	30	39	9
12	2	31	40	9
13	3	32	45	13
14	3	32	46	14
15	3	33	47	14
16	3	40	48	8

TABLE 15.7 Final Configuration Produced by Greedy Algorithm

Job Number	Family	Due Date	Completion Time	Lateness
1	1	5	5	0
2	1	6	6	0
3	1	10	7	−3
5	1	15	8	−7
4	2	13	13	0
6	2	15	14	−1
8	2	22	15	−7
7	1	22	20	−2
9	1	23	21	−2
11	2	30	26	−4
12	2	31	27	−4
10	3	29	32	3
13	3	32	33	1
14	3	32	34	2
15	3	33	35	2
16	3	40	36	−4

like this one tend to have many local optima. What we need, therefore, is a mechanism that will force the algorithm away from a local optimum in order to see if there are better sequences farther away.

Table 15.8 A Feasible Sequence

Job Number	Family	Due Date	Completion Time	Lateness
1	1	5	5	0
2	1	6	6	0
3	1	10	7	−3
5	1	15	8	−7
4	2	13	13	0
6	2	15	14	−1
8	2	22	15	−7
11	2	30	16	−14
12	2	31	17	−14
7	1	22	22	0
9	1	23	23	0
10	3	29	28	−1
13	3	32	29	3
14	3	32	30	−2
15	3	33	31	−2
16	3	40	32	−8

One way to do this is to prohibit (make "taboo") certain recently considered moves. This approach is called **tabu search** (see Glover 1990), and the list of recent (and now forbidden) moves is called a **tabu list.** In practice, there are many ways to characterize moves. One obvious (albeit inefficient) choice is the entire sequence. In this case, certain sequences would become tabu once they were evaluated. But because there are so *many* sequences, the tabu list would need to be very long to be effective. Another, more efficient but less precise, option is the location of the job in the sequence. Thus, the move placing job 4 after job 5 (as we did in our first move) would become tabu once it was considered the first time. But because we need only prohibit this move temporarily in order to prevent the algorithm from settling into a local minimum, the length of the tabu list is limited. Once a tabu move has been on the list long enough, it is discarded and can then be considered again.

The tabu search can be further refined by not considering moves that we know cannot make things better. For example, in the above problem we know that making the sequence anything but EDD *within* a family (i.e., between setups) will only make things worse. For example, we would never consider moving job 2 after job 1 since these are of the same family and job 1 has a due date that is earlier than that for job 2. This type of consideration can limit the number of moves that must be considered and therefore can speed the algorithm.

Although tabu search is simple in principle, its implementation can become complicated (see Woodruff and Spearman 1992 for a more detailed discussion). Also, there are many other heuristic approaches that can be applied to sequencing and scheduling problems. Researchers are continuing to evolve new methods and evaluate which work best for given problems. For more discussion on heuristic scheduling methods, see Morton and Pentico (1994) and Pinedo (1995).

15.4.3 Bottleneck Scheduling Results

An important conclusion of this section is that scheduling need not be as hopeless as a narrow interpretation of the complexity results from scheduling theory might suggest. By simplifying the environment (e.g., with CONWIP lines) and using well-chosen heuristics, managers can achieve reasonably effective scheduling procedures.

In pull systems, such as CONWIP lines, simple sequences are sufficient, since the timing of releases is controlled by progress of the system. If there are no setups, an EDD sequence is an appropriate choice for a single CONWIP line. It is also suitable for systems of CONWIP lines with shared resources, as long as there are no significant setups and the FISFO dispatching rule is used at the shared resources. If there are significant setups, then a simple sequence is still sufficient for CONWIP lines, but not an EDD one. However, practical heuristics, such as tabu search, can be used to find good solutions for this case.

15.5 Diagnostic Scheduling

Unfortunately, not all scheduling situations are amenable to simple bottleneck sequencing. In some systems, the identity of the bottleneck shifts, due to changes in the product mix—when different products have different process times on the machines—or capacities change frequently, perhaps as a result of a fluctuating labor force. In some factories, extremely complicated routings do not allow use of CONWIP or any other pull system. In still others, WIP in the system is reassigned to different customers in response to a constantly changing demand profile.

A glib suggestion for dealing with these situations is to get rid of them. In some systems where this is possible, it may be the most sensible course of action. However, in others it may actually be infeasible physically or economically. In such cases, most firms turn to some variant of MRP. In concept, MRP can be applied to almost any manufacturing environment. However, as we noted in Chapters 3 and 5, the basic MRP model is flawed because of its underlying assumptions, particularly that of infinite capacity. In response, production researchers and software vendors have devoted increasing attention to finite-capacity schedulers. As stated earlier, this approach is often too little, too late since it relies on the MRP release schedule as input. The goal of this section is to maintain the structure of the ERP hierarchy while removing the defect in the MRP scheduling model.

In the real world, effective scheduling is more than a matter of finding good solutions to mathematical problems. Two important considerations are the following:

1. *Models depend on data, which must be estimated.* A common parameter required by many scheduling models is a tardiness cost, which is used to make a tradeoff between customer service and inventory costs. However, almost no one we have encountered in industry is comfortable with specifying such a cost in advance of seeing its effect on the schedule.

2. *Many intangibles are not addressed by models.* Special customer considerations, changing shop floor conditions, evolving relationships with suppliers and subcontractors, and so forth make completely automatic scheduling all but impossible. Consequently, most scheduling professionals with whom we have spoken feel that an effective scheduling system must allow for human intervention. To make effective use of human intelligence, such a system should evaluate the *feasibility* (not optimality) of a given schedule and, if it is infeasible, suggest changes. Suggestions might include adding capacity via overtime, temporary workers, or subcontracting; pushing out due dates of certain jobs,

and splitting large jobs. Human judgment is required to choose wisely among these options, in order to address such questions as, Which customers will tolerate a late or partial shipment? Which parts can be subcontracted now? Which groups of workers can and cannot be asked to work overtime?

Neither optimization-based nor simulation-based approaches are well suited to evaluating candidate schedules and offering improvement alternatives. Perhaps because of this, a survey of scheduling software found no systems with more than trivial diagnostic capability (Arguello 1994).

In contrast, the ERP paradigm is intended to develop *and evaluate* production schedules. The master production schedule (MPS) provides the demand; material requirements planning (MRP) nets demand, determines material requirements, and offsets them to provide a release schedule; and capacity requirements planning checks the schedule for feasibility. As a planning framework, this is ideally suited to real-world production control. However, as we discussed earlier, the basic model in MRP is too simple to accurately represent what happens in the plant. Similarly CRP is an inaccurate check on MRP because it suffers from the same modeling flaw (fixed lead times) as MRP. Even if CRP were an accurate check on schedule feasibility, it does not offer useful diagnostics on how to correct infeasibilities.

Thus, our goal is to provide a scheduling process that preserves the appropriate ERP framework but eliminates the modeling flaws of MRP. In this section, we discuss how and why infeasibilities arise and then offer a procedure for detecting them and suggesting corrective measures.

15.5.1 Types of Schedule Infeasibility

There are two basic types of schedule infeasibility. **WIP infeasibility** is caused by inappropriate positioning of WIP. If there is insufficient WIP in the system to facilitate fulfillment of near term due dates, then the schedule will be infeasible regardless of the capacity. The only way to remedy a WIP infeasibility is to postpone (push out) demand. **Capacity infeasibility** is caused by having insufficient capacity. Capacity infeasibilities can be remedied by either pushing out demand or adding capacity.

Example:
We illustrate the types and effects of schedule infeasibility by considering a line with a demonstrated capacity of $r_b^P = 100$ units per day and a practical minimum process time of $T_0^P = 3$ days. Thus, by Little's Law, the average WIP level will be 300 units. Currently, there are 95 jobs that are expected to finish at the end of day 1; 90 that should finish by the end of day 2; and 115 that have just started. Of these last 115 jobs, 100 will finish at the end of day 3. The remaining 15 will finish on day 4 due to the capacity constraint. The demands, which start out low but increase to above capacity, are given in Table 15.9.

First observe that total demand for the first three days is 280 jobs, while there are 300 units of WIP and capacity (each job is one unit). Demand for the next 12 days is 1,190 units, while there is capacity to produce 1,200 over this interval plus 20 units of current WIP left over after filling demand for the first three days. Thus, from a quick aggregate perspective, meeting demand appears feasible.

However, when we look more closely, a problem becomes apparent. At the end of the first day the line will output 95 units to meet a demand of 90 units, which leaves five units of finished goods inventory (FGI). After the second day 90 additional units will be

TABLE 15.9 Demand for Diagnostics Example

Day from Start	Amount Due
1	90
2	100
3	90
4	80
5	70
6	130
7	120
8	110
9	110
10	110
11	100
12	90
13	90
14	90
15	90

output, but demand for that day is 100. Even after the five units of FGI left over from day 1 are used, this results in a deficit of five units. At the end of the third day 100 units are output to meet demand of 90 units, resulting in an excess of 10 units. This can cover the deficit from day 2, but only if we are willing to be a day late on delivery.

The reason for the deficit in day 2 is that there is not enough WIP in the system within two days of completion to cover demand during the first two days. While total demand for days 1 and 2 is $90 + 100 = 190$ units, there are only $95 + 90 = 185$ units of WIP that can be output by the end of day 2. Hence, a five-unit deficit will occur no matter how much capacity the line has. This is an example of a WIP infeasibility. Note that because it does not involve capacity, MRP can detect this type of infeasibility.

Looking at the demands beyond day 3, we see that there are other problems as well. Figure 15.5 shows the maximum cumulative production for the line relative to the cumulative demand for the line. Whenever maximum cumulative production falls below cumulative demand, the schedule is infeasible. The surplus line, whose scale is on the right, is the difference between the maximum cumulative production and the cumulative demand. Negative values indicate infeasibility. This curve first becomes negative in day 2—the infeasibility caused by insufficient WIP in the line. After that, the line can produce more than demand, and the surplus curve becomes positive. It becomes negative again on day 8 when demand begins to exceed capacity and stays negative until day 14 when the line finally catches back up.

The infeasibility in day 8 is different from that in day 2 because it *is* a function of capacity. While no amount of extra capacity could enable the line to meet demand in day 2, production of an additional 25 units of output sometime before day 8 would allow it to meet demand on that day. Hence the infeasibility that occurred on day 8 is an example of a capacity infeasibility. Because MRP and CRP are based on an infinite-capacity model, they cannot detect this type of infeasibility.

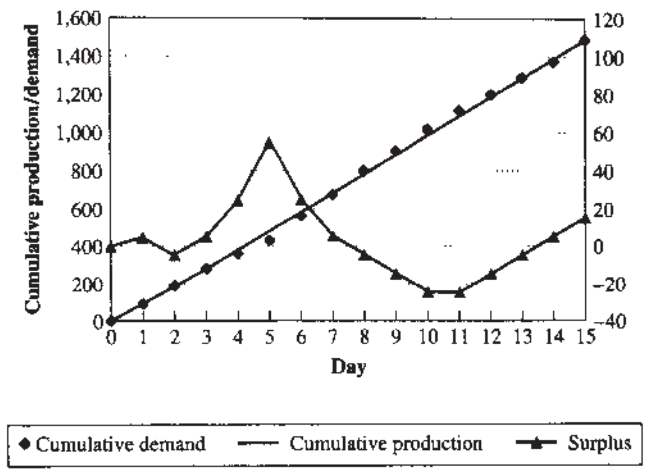


FIGURE 15.5

Demand versus available production and WIP

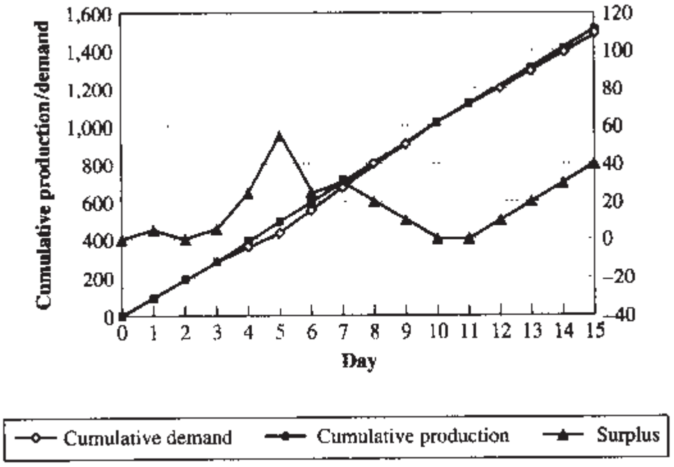


FIGURE 15.6

Demand versus available production and WIP after capacity increases

The two different types of infeasibilities require different remedies. Since adding capacity will not help a WIP infeasibility, the only solution is to push out due dates. For example, if five units of the 100 units due in day 2 could be pushed out to day 3, that portion of the schedule would become feasible.

Capacity infeasibilities can be remedied in two ways: by adding capacity or by pushing out due dates. For instance, if overtime were used on day 8 to produce 25 units of output, the schedule would be feasible. However, this will also increase the surplus by the end of the planning horizon (see Figure 15.6). Alternately, if 30 units of the 130 units demanded on day 6 are moved to days 12, 13, and 14 (10 each), the schedule also becomes feasible (see Figure 15.7). This results in a smaller surplus at the end of the planning horizon than occurs under the overtime alternative, since no capacity is added. Of course, in an actual scheduling situation we would have to correct these surpluses; the approach of the next section gives a procedure for doing this.

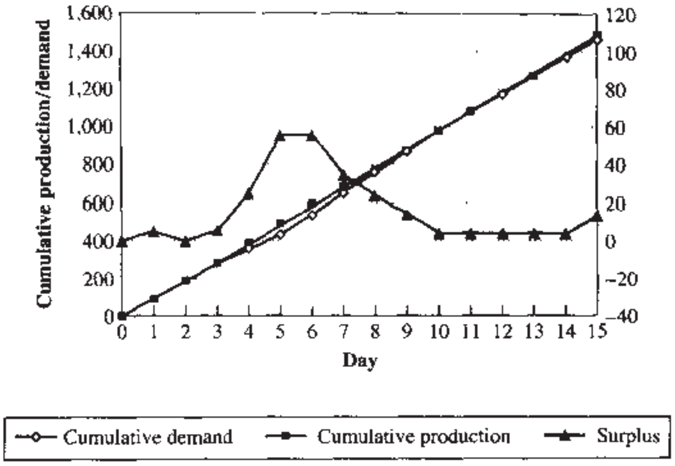


Figure 15.7

Demand versus available production and WIP after pushing out demand

15.5.2 Capacitated Material Requirements Planning—MRP-C

A procedure designed to detect and remedy scheduling infeasibilities is **capacitated material requirements planning** (**MRP-C**) (see Tardif 1995 for details). MRP-C is similar to MRP except that it explicitly considers capacity. As such, it replaces MRP in the MRP II planning hierarchy.

The basic structure of MRP-C derives from the hierarchical nature of production planning. As we saw in Chapter 13, decision variables in high-level problems are often constraints in lower-level problems. For example, aggregate planning may treat capacity variables (e.g., overtime) as variables. Demand management may treat customer due dates as variables (if due date quoting is used). But scheduling frequently treats both capacities and due dates as constraints. As we have seen, these constraints result in some of the hardest problems in the production research literature (e.g., the minimum makespan job shop problem).

To make the scheduling problem tractable, MRP-C begins by seeking a low-level schedule that satisfies all due dates without violating capacity and builds as little inventory as possible before it is needed. We define **build-ahead** as inventory that is made earlier than the actual demand. The objective of MRP-C is to find a minimum build-ahead feasible schedule. If it cannot find such a schedule, then MRP-C highlights the causes of the infeasibility, enabling the planner to make changes in due dates, capacities, or both as appropriate to the specific situation.

The algorithm used in MRP-C is based on the conveyor model to characterize the behavior of each process (machine, line segment, or line, depending on the level of detail required) in the system. This requires estimates of the following two parameters:

1. **Minimum practical lead time** is denoted by T_0^P and represents the time to go through the line with no queueing. This should include any common delays such as waiting to move and minor adjustments and so will usually be larger than the raw process time T_0 used in Chapter 7.
2. **Practical production rate** is denoted by $r_b^P(t)$ and represents the realistic capacity of the line in time period t. If $r_b^P(t)$ is constant for all t, then because utilization of the bottleneck must be less than 100 percent, $r_b^P(t)$ must be smaller than the bottleneck rate r_b used in Chapter 7. However, if $r_b^P(t)$ varies (e.g., due to scheduled overtime), then $r_b^P(t)$ may exceed r_b for some t. However, on average it must be smaller than the long-run bottleneck rate.

By using only two values, MRP-C captures the basic relationships between WIP and cycle time for an entire process without the burden of a full-blown simulation or detailed capacity knowledge of each station.

MRP-C consists of two phases. The first compares near-term demands against WIP already in the line and available capacity to see if there are any infeasibilities. Once all infeasibilities have been addressed, the second phase works backward in time to determine the minimum releases into the line required to meet demand. We describe the mechanics of phase 1 of MRP-C in the following technical note.

Technical Note—MRP-C (Phase I)

The procedures divides time into periods, which could represent shifts, days, or weeks, depending on the level of resolution desired, and makes use of the following notation:

T = planning horizon (last period) of problem

t = time index for periods, $t = 0, 1, \ldots, T$ (anything occurring in period 0 is before current period)

T_0^P = minimum practical lead time of process under consideration

ℓ = lead time to obtain raw material

$r_b^P(t)$ = production rate (capacity) of process in period t

$D(t)$ = demand due at time t, that is, the master production schedule

$a(t)$ = scheduled receipts (arrivals) of raw material in period t

$w(t)$ = "timed-available" WIP (TAWIP) in line that is t periods away from completion, defined for $t = 0, 1, \ldots, T_P$. Note that $w(0)$ represents WIP already completed, which could be finished goods inventory or raw material for another process, while $w(t)$ represents WIP that is t periods from completion.

$\hat{w}(t)$ = capacity-adjusted timed-available WIP (CATAWIP) that takes into consideration the amount of capacity available in period t

$c(t)$ = carryover WIP at t

$I(t)$ = projected-on-hand FGI at time t

$N(t)$ = net FGI requirements for period t

The first phase computes the net WIP requirements $N(t)$ as follows:

1. Determine TAWIP, which can be in the form of existing WIP in the line or scheduled receipts. We do this by setting

$$w(t) = \begin{cases} \text{existing WIP} & \text{for } 1 \le t \le T_0^P \\ a(t - T_0^P) & \text{for } T_0^P < t \le T_0^P + \ell \\ \infty & \text{for } t > T_0^P + \ell \end{cases}$$

For periods within the practical minimum process time, $w(t)$ is equal to the existing WIP in the line. In the previous example where $T_0^P = 3$ days, $w(1)$ has been in the line for two days and so is one day away from completion, $w(2)$ has been in for one day and therefore requires two more days for completion, and so on. For values of t beyond T_0^P but less than the time to obtain raw material, the timed-available WIP is equal to the arrivals of raw material received T_0^P periods before. For periods that are farther out than the raw material lead time (ℓ) plus the process time (T_0^P), the value is set to infinity since these materials can be ordered within their lead time.

2. Compute CATAWIP. We do this by starting with $c(0) = 0$ and computing

$$\hat{w}(t) = \min\{r_b^P(t), w(t) + c(t-1)\}$$

$$c(t) = w(t) + c(t-1) - \hat{w}(t)$$

for $t = 1, 2, \ldots, T$. This step accounts for the fact that no more than r_b^P units of WIP available in period t can actually be completed in period t, due to constrained capacity. So the $\hat{w}(t)$ values represent how much production can be done in each period running at full capacity. If more WIP is available than capacity in period t, then it is carried over as $c(t)$ and becomes available in period $t + 1$.

3. Compute projected on-hand FGI. We do this by starting with $I(0)$ equal to the initial finished goods inventory and computing

$$I(t) = I(t-1) + \hat{w}(t) - D(t)$$

for $t = 1, 2, \ldots, T$. Using the maximum available capacity/raw materials, this step computes the ending net FGI in each period. If this value ever becomes negative, then it means that there is not sufficient WIP and/or capacity to meet demand.

4. Compute the net requirements. We do this by computing

$$N(t) = \max\{0, \min\{-I(t), D(t)\}\}$$

for $t = 1, 2, \ldots, T$. If $I(t)$ is greater than zero, there are no net requirements because there is sufficient inventory to cover the gross requirements. If $I(t)$ is negative but $I(t-1) \geq 0$, then the net requirement is equal to the absolute value of $I(t)$. If $I(t)$ and $I(t-1)$ are both negative, then $N(t)$ will be equal to demand for the period. Note that this is exactly analogous to the netting calculation in regular MRP.

If $N(t) > 0$ and $c(t) < N(t)$, then the schedule is WIP-infeasible and the only remedy is to move out $N(t) - c(t)$ units of demand. If $N(t) > 0$ and $c(t) \geq N(t)$, then the problem is a capacity infeasibility, which can be remedied either by moving out demand or by adding capacity.

5. After any change is made (e.g., moving out a due date), all values must be recomputed.

The MRP-C procedure detailed above appears complex, but is actually very straightforward to implement in a spreadsheet. The following example gives an illustration.

Example:
Applying the MRP-C procedure to the data of the previous example generates the results shown in Table 15.10. The WIP infeasibility of five units in period 2 is indicated by the fact that $N(2) = 5$. The only way to address this problem is to reduce demand in period 2 from 100 to 95 and then to move it into period 3 by increasing demand from 90 to 95. The fact that $N(t)$ reaches 25 for $t = 10, 11$ indicates a shortage of 25 units of capacity. One way to address this problem is to add enough overtime to produce 25 more units in period 8 (which we do in Table 15.11). Otherwise, if no extra capacity is available, we could have postponed the production of 25 units to later in the schedule by pushing back due dates. The projected on-hand figure indicates periods with additional capacity and/or WIP that could accept extra demand. The final schedule is shown in Table 15.11.

At this point, we know that a feasible schedule exists. However, the master production schedule generated is not a good schedule since it has periods of demand that exceed capacity. Thus, some build-ahead of inventory must be done. The second phase of MRP-C uses the constraints of capacity and WIP provided by the first phase to compute a schedule that is feasible and produces a minimum of build-ahead inventory. This is done by computing the schedule from the last period and working backward in time. The procedure is given in the following technical note.

Table 15.10 Feasibility Calculations

Period t	Demand $D(t)$	TAWIP $w(t)$	Capacity $r_b^P(t)$	CATAWIP $\hat{w}(t)$	Carryover $c(t)$	Projected on Hand $I(t)$	Net Requirements $N(t)$
0					0	0	
1	90	95	100	95	0	5	0
2	100	90	100	90	0	−5	5
3	90	115	100	100	15	5	0
4	80	∞	100	100	∞	25	0
5	70	∞	100	100	∞	55	0
6	130	∞	100	100	∞	25	0
7	120	∞	100	100	∞	5	0
8	110	∞	100	100	∞	−5	5
9	110	∞	100	100	∞	−15	15
10	110	∞	100	100	∞	−25	25
11	100	∞	100	100	∞	−25	25
12	90	∞	100	100	∞	−15	15
13	90	∞	100	100	∞	−5	5
14	90	∞	100	100	∞	5	0
15	90	∞	100	100	∞	15	0

Table 15.11 Final Feasible Master Production Schedule

Period t	Demand $D(t)$	TAWIP $w(t)$	Capacity $r_b^P(t)$	CATAWIP $\hat{w}(t)$	Carryover $c(t)$	Projected on Hand $I(t)$	Net Requirements $N(t)$
0					0	0	
1	90	95	100	95	0	5	0
2	95	90	100	90	0	0	0
3	90	115	100	100	15	10	0
4	85	∞	100	100	∞	25	0
5	70	∞	100	100	∞	55	0
6	130	∞	100	100	∞	25	0
7	120	∞	100	100	∞	5	0
8	110	∞	125	125	∞	20	0
9	110	∞	100	100	∞	10	0
10	110	∞	100	100	∞	0	0
11	100	∞	100	100	∞	0	0
12	90	∞	100	100	∞	10	0
13	90	∞	100	100	∞	20	0
14	90	∞	100	100	∞	30	0
15	90	∞	100	100	∞	40	0

Technical Note—MRP-C (Phase II)

To describe the MRP-C procedure for converting a schedule of (feasible) demands to a schedule of releases (starts), we make use of the following notation:

$D(t)$ = demand due at time t, that is, master production schedule

$I(t)$ = projected on-hand FGI at time t

$N(t)$ = net FGI requirements for period t

$\hat{w}(t)$ = CATAWIP available in period t

$X(t)$ = production quantity in period t

$Y(t)$ = amount of build-ahead inventory in period t, which represents production in period t intended to fill demand in periods beyond t

$S(t)$ = release quantities ("starts") in period t

The basic procedure is to first compute net demand by subtracting finished goods inventory in much the same way as MRP. Then available production in each period is given by the capacity-adjusted timed-available WIP (CATAWIP). Since this includes WIP in the line, we do not net it out as we would do in MRP. With this, the procedure computes production, build-ahead, and starts for each period.

The specific steps are as follows:

1. *Netting.* We first compute net requirements in the standard (MRP) way.
   a. Initialize variables:

$$I(0) = \text{initial finished goods inventory}$$
$$N(0) = 0$$

   b. For each period, beginning with period 1 and working to period T, we compute the projected on-hand inventory and the net requirements as follows.

$$I(t) = I(t-1) + N(t-1) - D(t)$$
$$N(t) = \max\{0, \min\{D(t), -I(t)\}$$

2. *Scheduling.* The scheduling procedure is done from the last (T) period, working backward in time.
   a. Initialize variables.

$$D(T+1) = 0$$
$$X(T+1) = 0$$
$$Y(T+1) = \text{desired ending FGI level}$$

   b. For each period t, starting with T and working down to period 1, compute

$$Y(t) = Y(t+1) + D(t+1) - X(t+1)$$
$$X(t) = \min\{\hat{w}(t), D(t) + Y(t)\}$$

   c. The equation $Y(0) = Y(1) + D(1) - X(1)$ provides an easy capacity check. This value should be zero if all the infeasibilities were addressed in phase I. If not, the schedule is infeasible and phase I needs to be redone correctly.
   d. Assuming there are no remaining schedule infeasibilities, we compute the schedule of production starts by offsetting the production quantities by the minimum practical lead time as follows:

$$S(t) = X(t + T_0^P) \qquad \text{for } t = 1, 2, \ldots, T - T_0^P$$

The MRP-C scheduling procedure computes the amount of build-ahead from the end of the time horizon T backward. The level of build-ahead in period T is the desired level of inventory at the end of the planning horizon. One would generally set this to zero, unless there were some exceptional reason to plan to finish the planning horizon with excess inventory. At each period, output will be either the capacity or the total demand (net demand plus build-ahead), whichever is less. This is intuitive since production *cannot* exceed the maximum rate of the line and *should not* exceed demand (including build-ahead).

If the build-ahead for period 0 is positive, the schedule is infeasible. The amount of build-ahead in period 0 indicates the amount of additional finished inventory needed at $t = 0$ to make the schedule feasible. However, if phase I has addressed all the capacity and WIP infeasibilities, $Y(0)$ will be zero. Indeed, this is the entire point of phase I.

The final output of the MRP-C procedure is a list of production starts that will meet all the (possibly revised) due dates within capacity and material constraints while producing a minimum of build-ahead inventory.

Example:
We continue with our example from phase I and apply the second phase of MRP-C. This generates the results in Table 15.12. Note that the schedule calls for production to be as high as possible, being limited by WIP in the first two periods, and then limited by capacity thereafter, until period 12. At this point, production decreases to 90 units, which is below CATAWIP but is sufficient to keep up with demand.

Notice that while MRP-C does the dirty work of finding infeasibilities and identifying possible actions for remedying them, it leaves the sensitive judgments concerning increasing capacity (whether, how, where) and delaying jobs (which ones, how much) up to the user. As such, MRP-C encourages appropriate use of the respective talents of humans and computers in the scheduling process.

TABLE 15.12 Final Production Schedule

Period t	Demand $D(t)$	Projected on Hand $I(t)$	Net Requirements $N(t)$	CATAWIP $\hat{w}(t)$	Build-Ahead $Y(t)$	Production $X(t)$	Starts $S(t)$
0		0	0		0		
1	90	−90	90	95	5	95	100
2	100	−95	95	90	0	90	100
3	90	−95	95	100	5	100	100
4	80	−80	80	100	25	100	100
5	70	−70	70	100	55	100	125
6	130	−130	130	100	25	100	100
7	120	−120	120	100	5	100	100
8	110	−110	110	125	20	125	100
9	110	−110	110	100	10	100	90
10	110	−110	110	100	0	100	90
11	100	−100	100	100	0	100	90
12	90	−90	90	100	0	90	90
13	90	−90	90	100	0	90	
14	90	−90	90	100	0	90	
15	90	−90	90	100	0	90	

15.5.3 Extending MRP-C to More General Environments

The preceding described how to use the MRP-C procedure to schedule one process (workstation, line, or line segment) represented by the conveyor model. The real power of MRP-C is that it can be extended to multistage systems with more than a single product.

For a serial line, this extension is simple. The production starts into a downstream station represent the demands upon the upstream station that feeds it. Thus, we can simply apply MRP-C by starting at the last station and working backward to the front of the line. Likewise, the time-adjusted WIP (TAWIP) levels will be generated by the production of the upstream process.

If there are assembly stations, then production starts must be translated to demands upon each of the stations feeding them. This is exactly analogous to the bill-of-material explosion concept of MRP, except applied to routings. Otherwise the MRP-C procedure remains unchanged.

In systems where multiple routings (i.e., producing different products) pass through a single station, we must combine the individual demands (i.e., production starts at downstream stations) to form *aggregate* demand. Since the different products may have different processing times at the shared resource, it is important that the MRP-C calculations be done in units of time instead of product. That is, capacity, demand, WIP, and so forth should all be measured in hours. This is similar in spirit to the idea of maintaining a constant amount of work rather than a constant number of units in a CONWIP line with multiple products, which we discussed in Chapter 14.

In systems with multiple products, things get a bit more complex because we must choose a method for breaking ties when more than one product requires build-ahead in the same period. The wrong choice can schedule early production of a product with little or no available WIP instead of another product that has plentiful WIP. This can cause a WIP infeasibility when the next stage is scheduled. Several clever means for breaking ties have been proposed by Tardif (1995), who also addresses other practical implementation issues.

15.5.4 Practical Issues

The MRP-C approach has two clear advantages over MRP: (1) It uses a more accurate model that explicitly considers capacity, and (2) it provides the planner with useful diagnostics. However, there are some problems.

First, MRP-C relies on a heuristic and therefore cannot be guaranteed to find a feasible schedule if one exists. (However, if it finds a feasible schedule, this schedule *is* truly feasible.) Although certain cases of MRP-C can make use of an exact algorithm, this is much slower (see Tardif 1995). In essence, the approach discussed above sacrifices accuracy for speed. Given that it is intended for use in an iterative, "decision support" mode, the additional speed is probably worth the small sacrifice in accuracy. Moreover, any errors produced by MRP-C will make the schedule more conservative. That is, MRP-C may require more adjustments than the minimum necessary to achieve feasibility. Hence, schedules will be "more feasible" than they really need to be and will thus have a better chance of being successfully executed.

Second, MRP-C, like virtually all scheduling approaches, implies a *push* philosophy (i.e., it sets release *times*). As we discussed in Chapter 10, this makes it subject to all the drawbacks of push systems. Fortunately, one can integrate MRP-C (and indeed any push system, including MRP) into a pull environment and obtain many of the efficiency, predictability, and robustness benefits associated with pull. We describe how this can be done in the following section.

15.6 Production Scheduling in a Pull Environment

Recall the definitions of push and pull production control. A push system *schedules* releases into the line based on due dates, while a pull system *authorizes* releases into the line based on operating conditions. Push systems control release rates (and thereby throughput) and measure WIP to see if the rates are too large or too small. Pull systems do the opposite. They control WIP and measure completions to determine whether production is adequate. Since WIP control is less sensitive than release control, pull systems are more robust to errors than are push systems. Also, since pull systems directly control WIP, they avoid WIP explosions and the associated overtime vicious cycle often observed in push systems. Finally, pull systems have the ability to work ahead for short periods, allowing them to exploit periods of better-than-average production.

For these reasons, we want to maintain the benefits of pull systems to whatever extent possible. The question is, How can it be done in an environment that requires a detailed schedule? In this section we discuss the link between scheduling and pull production.

15.6.1 Schedule Planning, Pull Execution

Even the best schedule is only a plan of what should happen, not a guarantee of what will happen. By necessity, schedules are prepared relatively infrequently compared to shop floor activity; the schedule may be regenerated weekly, while material flow, machine failures, and so forth happen in real time. Hence, they cannot help but become outdated, sometimes very rapidly. Therefore we should treat the schedule as a set of suggestions, not a set of requirements, concerning the order and timing of releases into the system.

A pull system is an ideal mechanism for linking releases to real-time status information. When the line is already congested with WIP, so that further releases will only increase congestion without making jobs finish sooner, a pull system will prevent releases. When the line runs faster than expected and has capacity for more work, a pull system will draw it in. Fortunately, using a pull system in concert with a schedule is not at all difficult.

To illustrate how this would work, suppose we have a CONWIP system in place for each routing and make use of MRP-C to generate a schedule for the overall system. Note that there is an important link between MRP-C and CONWIP: the conveyor model. Thus, if the parameters are correct, MRP-C will generate a set of release times that are very close to the times that the CONWIP system generates authorizations (pull signals) for the releases. Of course, variability will always prevent a perfect match, but on average actual performance will be consistent with the planned schedule.

When production falls behind schedule, we can catch up if there is a capacity cushion (e.g., a makeup time at the end of each shift or day) available. If no such cushion is available, we must adjust the schedule at the next regeneration. When production outpaces the schedule, we can allow it to work ahead, by allowing the line to pull in more than was planned. A simple rule comparing the current date and time with the date and time of the next release can keep the CONWIP line from working too far ahead. In this way, the CONWIP system can take advantage of the "good" production days without getting too far from schedule.

When we cannot rely on a capacity cushion to make up for lags in production (e.g., we are running the line as fast as we can), we can supplement the CONWIP control system with the statistical throughput control (STC) procedure described in Chapter 13. This provides a means for detecting when production is out of control relative to the schedule. When this occurs, either the system or the MRP-C parameters need adjustment. Which to adjust may pose an important management decision. Reducing MRP-C capacity parameters may be tantamount to admitting that corporate goals are not achievable. However, increasing capacity may require investment in equipment, staff, increased subcontracting costs, or consulting.

15.6.2 Using CONWIP with MRP

Nothing in the previous discussion about using CONWIP in conjunction with a schedule absolutely requires that the schedule be generated with MRP-C. Of course, since MRP-C considers capacity using the same conveyor model that underlies CONWIP, we would expect it to work well. But we can certainly use CONWIP with *any* scheduling system, including MRP. We would do this by using the MRP-generated list of planned order releases, sorted by routing, as the work backlogs for each CONWIP line. The CONWIP system then determines when jobs actually get pulled into the system.

As with MRP-C, we can employ a capacity cushion, work ahead, and track against schedule. The primary difference is that the underlying model of MRP and CONWIP are *not* consistent. Consequently, MRP is more likely to generate inconsistent planned order release schedules than is MRP-C. This can be mitigated, somewhat, by employing good master production scheduling techniques and by debugging the process using bottom-up replanning.

15.7 Conclusions

Production problems are notoriously difficult, both because they involve many conflicting goals and because the underlying mathematics can get very complex. Considerable scheduling research has produced formalized measures of the complexity of scheduling problems and has generated some good insights. However, it has not yielded good solutions to practical scheduling situations.

Because scheduling is difficult, an important insight of our discussion is that it is frequently possible to avoid hard problems by solving different ones. One example is to replace a system of exogenously generated due dates with a systematic means for quoting them. Another is to separate the problem of keeping cycle times short (solve by using small jobs) from the problem of keeping capacities high (solve by sequencing like jobs together for fewer setups). Given an appropriately formulated problem, good heuristics for identifying feasible (not optimal) schedules are becoming available.

An important recent trend in scheduling research and software development is toward finite-capacity scheduling. By overcoming the fundamental flaw in MRP, these models have the potential to make the MRP II hierarchy much more effective in practice. However, to provide flexibility for accommodating intangibles, an effective approach to finite-capacity scheduling is for the system to evaluate schedule feasibility and generate diagnostics about infeasibilities. A procedure designed to do this is capacitated material requirements planning—MRP-C.

Finally, although scheduling is essentially a push philosophy, it is possible to use a schedule in concert with a pull system. The basic idea is to use the schedule to plan work releases and the pull system to execute them. This offers the planning benefits of a scheduling system along with the environmental benefits of a pull system.

Study Questions

1. What are some goals of production scheduling? How do these conflict?
2. How does reducing cycle time support several of the above goals?
3. What motivates maximizing utilization? What motivates not maximizing utilization?
4. Why is average tardiness a better measure than average lateness?
5. What are some drawbacks of using service level as the only measure of due date performance?
6. For each of the assumptions of classic scheduling theory, give an example of when it might be valid. Give an example of when each is not valid.
7. Why do people use dispatching rules instead of finding an optimal schedule?
8. What dispatching rule minimizes average cycle time for a deterministic single machine? What rule minimizes maximum tardiness? How can one easily check to see if a schedule exists for which there are no tardy jobs?
9. Provide an argument that no matter how sophisticated the dispatching rule, it cannot solve the problem of minimizing average tardiness.
10. What is some evidence that there are some scheduling problems for which no polynomial algorithm exists?
11. Address the following comment: "Well, maybe today's computers are too slow to solve the job shop scheduling problem, but new parallel processing technology will speed them up to the point where computer time should not be an obstacle to solving it in the near future."
12. What higher-level planning problems are related to the production scheduling problem? What are the variables and constraints in the high-level problems? What are the variables and constraints in the lower-level scheduling problem? How are the problems linked?
13. How well do you think the policy of planning with a schedule and executing with a pull system should work using MRP-C and CONWIP? Why? How well should it work using MRP and kanban? Why?

Problems

1. Consider the following three jobs to be processed on a single machine:

Job Number	Process Time	Due Date
1	4	2
2	2	3
3	1	4

   Enumerate all possible sequences and compute the average cycle time, total tardiness, and maximum lateness for each. Which sequence works best for each measure? Identify it as EDD, SPT, or something else.

2. You are in charge of the shearing and pressing operations in a job shop. When you arrived this morning, there were seven jobs with the following processing times.

	Processing Time	
Job	Shear	Press
1	6	3
2	2	9
3	5	3
4	1	8
5	7	1
6	4	5
7	9	6

a. What is the makespan under the SPT dispatching rule?

b. What sequence yields the minimum makespan?

c. What is this makespan?

3. Your boss knows factory physics and insists on reducing average cycle time to help keep jobs on time and reduce congestion. For this reason, your personal performance evaluation is based on the average cycle time of the jobs through your process center. However, your boss also knows that late jobs are *extremely bad,* and she will fire you if you produce a schedule that includes any late jobs. The jobs listed below are staged in your process center for the first shift. Sequence them such that your evaluation will be the best it can be without getting you fired.

	Job				
	J_1	J_2	J_3	J_4	J_5
Processing time	6	2	4	9	3
Due date	33	13	6	23	31

4. Suppose daily production of a CONWIP line is nearly normally distributed with a mean of 250 pieces and a standard deviation of 50 pieces. The WIP level of the CONWIP line is 1,250 pieces. Currently there is a backlog of 1,400 pieces with an "emergency position" 150 pieces out. A new order for 100 pieces arrives.
   a. Quote a lead time with 95 percent confidence if the new order is placed at the end of the backlog and if it is placed in the emergency position.
   b. Quote a lead time with 99 percent confidence if the new order is placed at the end of the backlog and if it is placed in the emergency position.
5. Consider the jobs on the next page. Process times for all jobs are one hour. Changeovers between families require four hours. Thus, the completion time for job 1 is 5, for job 2 is 6, for job 3 is 11, and so on.

Job	Family Code	Due Date
1	1	5
2	1	6
3	2	12
4	2	13
5	1	13
6	1	19
7	1	20
8	2	20
9	2	26
10	1	28

a. Compute the total tardiness of the sequence.

b. How many possible sequences are there?

c. Find a sequence with no tardiness.

6. The Hickory Flat Sawmill (HFS) makes four kinds of lumber in one mill. Orders come from a variety of lumber companies to a central warehouse. Whenever the warehouse hits the reorder point, an order is placed to HFS. Pappy Red, the sawmill manager, has set the lot sizes to be run on the mill based on historical demands and common sense. The smallest amount made is a lot of 1,000 board-feet (1 kbf). The time it takes to process a lot depends on the product, but the time does not vary more than 25 percent from the mean. The changeover time can be quite long depending on how long it takes to get the mill producing good product again. The shortest time that anyone can remember is two hours. Once it took all day (eight hours). Most of the time it takes around four hours. Demand data and run rates are given in Table 15.13. The mill runs productively eight hours per day, five days per week (assume 4.33 weeks per month).

   The lot sizes are 50 of the knotty 1 × 10, 34 for the clear 1 × 4, 45 for the clear 1 × 6, and 40 for the rough plank. Lots are run on a first-come, first-served basis as they arrive from the warehouse. Currently the average response time is nearly three weeks (14.3 working days). The distributor has told HFS that HFS needs to get this down to two weeks in order to continue being a supplier.

   a. Compute the effective SCV c_e^2 for the mill. What portion of c_e^2 is due to the term in square brackets in Equation (15.8)? What can you do to reduce it?

   b. Verify the 14.3-working-day cycle time.

   c. What can you do to reduce cycle times without investing in any more equipment or physical process improvements?

TABLE 15.13 Data for the Sawmill Problem

Parameter	Knotty 1 x 10	Clear 1 x 4	Clear 1 x 6	Rough Plank
Demand (kbf/mo)	50	170	45	80
One lot time (hour)	0.2000	0.4000	0.6000	0.1000

7. Single parts arrive to a furnace at a rate of 100 per hour with exponential times between arrivals. The furnace time is three hours with essentially no variability. It can hold 500 parts. Find the batch size that minimizes total cycle time at the furnace.

8. Consider a serial line composed of three workstations. The first workstation has a production rate of 100 units per day and a minimum practical lead time T_0^P of three days. The second has a rate of 90 units per day and $T_0^P = 4$ days; and the third has a rate of 100 and $T_0^P = 3$ days. Lead time for raw material is one day, and there are currently 100 units on hand.

   Currently there are 450 units of finished goods, 95 units ready to go into finished goods on the first day, 95 on the second, and 100 on the third; all from the last station. The middle station has 35 units completed and ready to move to the last station and 90 units ready to come out in each of the next four days. The first station has no WIP completed, 95 units that will finish on the first day, zero units that will finish the second day, and 100 units that will finish the third day.

   The demand for the line is given in the table below.

Day from Start	Amount Due
1	80
2	80
3	80
4	80
5	80
6	130
7	150
8	180
9	220
10	240
11	210
12	150
13	90
14	80
15	80

Develop a feasible schedule that minimizes the amount of inventory required. If it is infeasible, adjust demands by moving them out. However, all demand must be met within 17 days.

CHAPTER

16 Aggregate and Workforce Planning

And I remember misinformation followed us like a plague,
Nobody knew from time to time if the plans were changed.
Paul Simon

16.1 Introduction

A variety of manufacturing management decisions require information about what a plant will produce over the next year or two. Exampes include the following:

1. *Staffing.* Recruiting and training new workers is a time-consuming process. Management needs a long-term production plan to decide how many and what type of workers to add and when to bring them on-line in order to meet production needs. Conversely, eliminating workers is costly and painful, but sometimes necessary. Anticipating reductions via a long-term plan makes it possible to use natural attrition, or other gentler methods, in place of layoffs to achieve at least part of the reductions.
2. *Procurement.* Contracts with suppliers are frequently set up well in advance of placing actual orders. For example, a firm might need an opportunity to "certify" the subcontractor for quality and other performance measures. Additionally, some procurement lead times are long (e.g., for high-technology components they may be six months or more). Therefore, decisions regarding contracts and long-lead-time orders must be made on the basis of a long-term production plan.
3. *Subcontracting.* Management must arrange contracts with subcontractors to manufacture entire components or to perform specific operations well in advance of actually sending out orders. Determining what types of subcontracting to use requires long-term projections of production requirements and a plan for in-house capacity modifications.
4. *Marketing.* Marketing personnel should make decisions on which products to promote on the basis of both a demand forecast *and* knowledge of which products have tight capacity and which do not. A long-term production plan incorporating planned capacity changes is needed for this.

The module in which we address the important question of what will be produced and when it will be produced over the long range is the **aggregate planning (AP)** module. As Figure 13.2 illustrated, the AP module occupies a central position in the production planning and control (PPC) hierarchy. The reason, or course, is that so many important decisions, such as those listed, depend on a long-term production plan.

Precisely because so many different decisions hinge on the long-range production plan, many different formulations of the AP module are possible. Which formulation is appropriate depends on what decision is being addressed. A model for determining the time of staffing additions may be very different from a model for deciding which products should be manufactured by outside subcontractors. Yet a different model might make sense if we want to address both issues simultaneously.

The staffing problem is of sufficient importance to warrant its own module in the hierarchy of Figure 13.2, the **workforce planning (WP)** module. Although high-level workforce planning (projections of total staffing increases or decreases, institution of training policies) can be done using only a rough estimate of future production based on the demand forecast, low-level staffing decisions (timing of hires or layoffs, scheduling usage of temporary hires, scheduling training) are often based on the more detailed production information contained in the aggregate plan. In the context of the PPC hierarchy in Figure 13.2, we can think of the AP module as either refining the output of the WP module or working in concert with the WP module. In any case, they are closely related. We highlight this relationship by treating aggregate planning and workforce planning together in this chapter.

As we mentioned in Chapter 13, linear programming is a particularly useful tool for formulating and solving many of the problems commonly faced in the aggregate planning and workforce planning modules. In this chapter, we will formulate several typical AP/WP problems as linear programs (LPs). We will also demonstrate the use of linear programming (LP) as a solution tool in various examples. Our goal is not so much to provide specific solutions to particular AP programs, but rather to illustrate general problem-solving approaches. The reader should be able to combine and extend our solutions to cover situations not directly addressed here.

Finally, while this chapter will not make an LP expert out of the reader, we do hope that he or she will become aware of how and where LP can be used in solving AP problems. If managers can recognize that particular problems are well suited to LP, they can easily obtain the technical support (consultants, internal experts) for carrying out the analysis and implementation. Unfortunately, far too few practicing managers make this connection; as a result, many are hammering away at problems that are well suited to linear programming with manual spreadsheets and other ad hoc approaches.

16.2 Basic Aggregate Planning

We start with a discussion of simple aggregate planning situations and work our way up to more complex cases. Throughout the chapter, we assume that we have **a demand forecast** available to us. This forecast is generated by the forecasting module and gives estimates of periodic demand over the **planning horizon.** Typically, periods are given in months, although further into the future they can represent longer intervals. For instance, periods 1 to 12 might represent the next 12 months, while periods 13 to 16 might represent the four quarters following these 12 months. A typical planning horizon for an AP module is one to three years.

16.2.1 A Simple Model

Our first scenario represents the simplest possible AP module. We consider this case not because it leads to a practical model, but because it illustrates the basic issues, provides a basis for considering more realistic situations, and showcases how linear programming can support the aggregate planning process. Although our discussion does not presume any background in linear programming, the reader interested in how and why LP works is advised to consult Appendix 16A, which provides an elementary overview of this important technique.

For modeling purposes, we consider the situation where there is only a single product, and the entire plant can be treated as a single resource. In every period, we have a demand forecast and a capacity constraint. For simplicity, we assume that demands represent customer orders that are due at the end of the period, and we neglect randomness and yield loss.

It is obvious under these simplifying assumptions that if demand is less than capacity in every period, the optimal solution is to simply produce amounts equal to demand in every period. This solution will meet all demand just-in-time and therefore will not build up any inventory between periods. However, if demand exceeds capacity in some periods, then we must work ahead (i.e., produce more than we need in some previous period). If demand cannot be met even by working ahead, we want our model to tell us this. To model this situation in the form of a linear program, we introduce the following notation:

t = an index of time periods, where $t = 1, \ldots, \bar{t}$, so $\bar{t}$ is planning horizon for problem

d_t = demand in period t, in physical units, standard containers, or some other appropriate quantity (assumed due at end of period)

c_t = capacity in period t, in same units used for d_t

r = profit per unit of product sold (not including inventory-carrying cost)

h = cost to hold one unit of inventory for one period

X_t = quantity produced during period t (assumed available to satisfy demand at end of period t)

S_t = quantity sold during period t (we assume that units produced in t are available for sale in t and thereafter)

I_t = inventory at end of period t (after demand has been met); we assume I_0 is given as data

In this notation, X_t, S_t, and I_t are **decision variables.** That is, the computer program solving the LP is free to choose their values so as to minimize the objective, provided the constraints are satisfied. The other variables—d_t, c_t, r, h—are **constants,** which must be estimated for the actual system and supplied as data. Throughout this chapter, we use the convention of representing variables with capital letters and constants with lowercase letters.

We can represent the problem of maximizing net profit minus inventory carrying cost subject to capacity and demand constraints as

$$\text{Maximize} \quad \sum_{t=1}^{\bar{t}} rS_t - hI_t \tag{16.1}$$

Subject to:

$$S_t \le d_t \qquad t = 1, \ldots, \bar{t} \tag{16.2}$$

$$X_t \le c_t \qquad t = 1, \ldots, \bar{t} \tag{16.3}$$

$$I_t = I_{t-1} + X_t - S_t \qquad t = 1, \ldots, \bar{t} \tag{16.4}$$

$$X_t, S_t, I_t \ge 0 \qquad t = 1, \ldots, \bar{t} \tag{16.5}$$

The objective function computes net profit by multiplying unit profit r by sales S_t in each period t, and subtracting the inventory carrying cost h times remaining inventory I_t at the end of period t, and summing over all periods in the planning horizon. Constraints (16.2) limit sales to demand. If possible, the computer will make all these constraints tight, since increasing the S_t values increases the objective function. The only reason that these constraints will not be tight in the optimal solution is that capacity constraints (16.3) will not permit it.[1] Constraints (16.4), which are of a form common to almost all multiperiod aggregate planning models, are known as **balance constraints.** Physically, all they represent is conservation of material; the inventory at the end of period $t\,(I_t)$ is equal to the inventory at the end of period $t - 1(I_{t-1})$ plus what was produced during period $t\,(X_t)$ minus the amount sold in period $t\ (S_t)$. These constraints are what force the computer to choose values for X_t, S_t, and I_t that are consistent with our verbal definitions of them. Constraints (16.5) are simple nonnegativity constraints, which rule out negative production or inventory levels. Many, but not all, computer packages for solving LPs automatically force decision variables to be nonnegative unless the user specifies otherwise.

16.2.2 An LP Example

To make the above formulation concrete and to illustrate the mechanics of solving it via linear programming, we now consider a simple example. The Excel spreadsheet shown in Figure 16.1 contains the unit profit r of $10, the one-period unit holding cost h of $1, the initial inventory I_0 of 0, and capacity and demand data c_t and d_t for the next six months. We will make use of the rest of the spreadsheet in Figure 16.1 momentarily. For now, we can express LP (16.1)–(16.5) for this specific case as

$$\text{Maximize}\quad 10(S_1 + S_2 + S_3 + S_4 + S_5 + S_6) - 1(I_1 + I_2 + I_3 + I_4 + I_5 + I_6) \tag{16.6}$$

Subject to:

Demand constraints

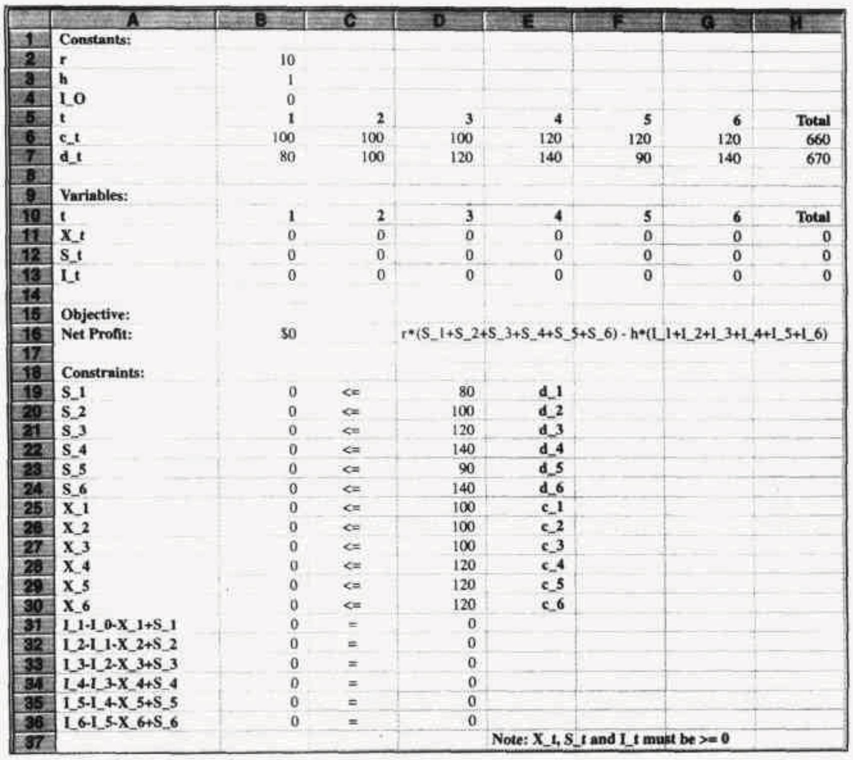
$$S_1 \le 80 \tag{16.7}$$
$$S_2 \le 100 \tag{16.8}$$
$$S_3 \le 120 \tag{16.9}$$
$$S_4 \le 140 \tag{16.10}$$
$$S_5 \le 90 \tag{16.11}$$
$$S_6 \le 140 \tag{16.12}$$

Capacity constraints

$$X_1 \le 100 \tag{16.13}$$
$$X_2 \le 100 \tag{16.14}$$
$$X_3 \le 100 \tag{16.15}$$
$$X_4 \le 120 \tag{16.16}$$
$$X_5 \le 120 \tag{16.17}$$
$$X_6 \le 120 \tag{16.18}$$

[1] If we want to consider demand as inviolable, we could remove constraints (16.2) and replace S_t with d_t in the objective and constraints (16.4). The problem with this, however, is that if demand is capacity-infeasible, the computer will just come back with a message saying "infeasible," which doesn't tell us why. The formulation here will be feasible regardless of demand; it simply won't make sales equal to demand if there is not enough capacity, and thus we will know what demand we are incapable of meeting from the solution.

FIGURE 16.1
Input spreadsheet for linear programming example



	A	B	C	D	E	F	G	H
1	Constants:							
2	r	10						
3	h	1						
4	I_O	0						
5	t	1	2	3	4	5	6	Total
6	c_t	100	100	100	120	120	120	660
7	d_t	80	100	120	140	90	140	670
8								
9	Variables:							
10	t	1	2	3	4	5	6	Total
11	X_t	0	0	0	0	0	0	0
12	S_t	0	0	0	0	0	0	0
13	I_t	0	0	0	0	0	0	0
14								
15	Objective:							
16	Net Profit:	$0		r*(S_1+S_2+S_3+S_4+S_5+S_6) - h*(I_1+I_2+I_3+I_4+I_5+I_6)				
17								
18	Constraints:							
19	S_1	0	<=	80	d_1			
20	S_2	0	<=	100	d_2			
21	S_3	0	<=	120	d_3			
22	S_4	0	<=	140	d_4			
23	S_5	0	<=	90	d_5			
24	S_6	0	<=	140	d_6			
25	X_1	0	<=	100	c_1			
26	X_2	0	<=	100	c_2			
27	X_3	0	<=	100	c_3			
28	X_4	0	<=	120	c_4			
29	X_5	0	<=	120	c_5			
30	X_6	0	<=	120	c_6			
31	I_1-I_0-X_1+S_1	0	=	0				
32	I_2-I_1-X_2+S_2	0	=	0				
33	I_3-I_2-X_3+S_3	0	=	0				
34	I_4-I_3-X_4+S_4	0	=	0				
35	I_5-I_4-X_5+S_5	0	=	0				
36	I_6-I_5-X_6+S_6	0	=	0				
37					Note: X_t, S_t and I_t must be >= 0			

Inventory balance constraints

$$I_1 - X_1 + S_1 = 0 \tag{16.19}$$

$$I_2 - I_1 - X_2 + S_2 = 0 \tag{16.20}$$

$$I_3 - I_2 - X_3 + S_3 = 0 \tag{16.21}$$

$$I_4 - I_3 - X_4 + S_4 = 0 \tag{16.22}$$

$$I_5 - I_4 - X_5 + S_5 = 0 \tag{16.23}$$

$$I_6 - I_5 - X_6 + S_6 = 0 \tag{16.24}$$

Nonnegativity constraints

$$X_1, X_2, X_3, X_4, X_5, X_6 \geq 0 \tag{16.25}$$

$$S_1, S_2, S_3, S_4, S_5, S_6 \geq 0 \tag{16.26}$$

$$I_1, I_2, I_3, I_4, I_5, I_6 \geq 0 \tag{16.27}$$

Some linear programming packages allow entry of a problem formulation in a format almost identical to (16.6) to (16.27) via a text editor. While this is certainly convenient for very small problems, it can become prohibitively tedious for large ones. Because of this, there is considerable work going on in the OM research community to develop **modeling languages** that provide user-friendly interfaces for describing large-scale optimization problems (see Fourer, Gay, and Kernighan 1993 for an excellent example of a modeling language). Conveniently for us, LP is becoming so prevalent that our spreadsheet package, Microsoft Excel, has an LP solver built right into it. We can represent and solve formulation (16.6) to (16.27) right in the spreadsheet shown in Figure 16.1. The following technical note provides details on how to do this.

Technical Note—Using the Excel LP Solver

Although the reader should consult the Excel documentation for details about the release in use, we will provide a brief overview of the LP solver in Excel 5.0. The first step is to establish cells for the decision variables (B11:G13 in Figure 16.1). We have initially entered zeros for these, but we can set them to be anything we like; thus, we could start by setting $X_t = d_t$, which would be closer to an optimal solution than zeros. The spreadsheet is a good place to play what-if games with the data. However, eventually we will turn over the problem of finding optimal values for the decision variables to the LP solver. Notice that for convenience we have also entered a column that totals X_t, S_t, and I_t. For example, cell H11 contains a formula to sum cells B11:G11. This allows us to write the objective function more compactly.

Once we have specified decision variables, we construct an objective function in cell B16. We do this by writing a formula that multiplies r (cell B2) by total sales (cell H12) and then subtracts the product of h (cell B3) and total inventory (cell H13). Since all the decision variables are zero at present, this formula also returns a zero; that is, the net profit on no production with no initial inventory is zero.

Next we need to specify the constraints (16.7) to (16.27). To do this, we need to develop formulas that compute the left-hand side of each constraint. For constraints (16.7) to (16.18) we really do not need to do this, since the left-hand sides are only X_t and S_t and we already have cells for these in the variables portion of the spreadsheet. However, for clarity, we will copy them to cells B19:B30. We will not do the same for the nonnegativity constraints (16.25) to (16.27), since it is a simple matter to choose all the decision variables and force them to be greater than or equal to zero in the Excel Solver menu. Constraints (16.19) to (16.24) require us to do work, since the left-hand sides are formulas of multiple variables. For instance, cell B31 contains a formula to compute $I_1 - I_0 - X_1 + S_1$ (that is, B13 − B4 − B11 + B12). We have given these cells names to remind us of what they represent, although any names could be used, since they are not necessary for the computation. We have also copied the values of the right-hand sides of the constraints into cells D19:D36 and labeled them in column E for clarity. This is not strictly necessary, but does make it easier to specify constraints in the Excel Solver, since whole blocks of constraints can be specified (for example, B19:B30 ≤ D19:D30). The equality and inequality symbols in column C are also unnecessary, but make the formulation easier to read.

To use the Excel LP Solver, we choose **Formula/Solver** from the menu. In the dialog box that comes up (see Figure 16.2), we specify the cells containing the objective, choose to maximize or minimize, and specify the cells containing decision variables (this can be done by pointing with the mouse). Then we add constraints by choosing **Add** from the constraints section of the form. Another dialog box (see Figure 16.3) comes up in which we fill in the cell containing the left-hand side of the constraint, choose the relationship (≥, ≤, or =), and fill in the right-hand side.

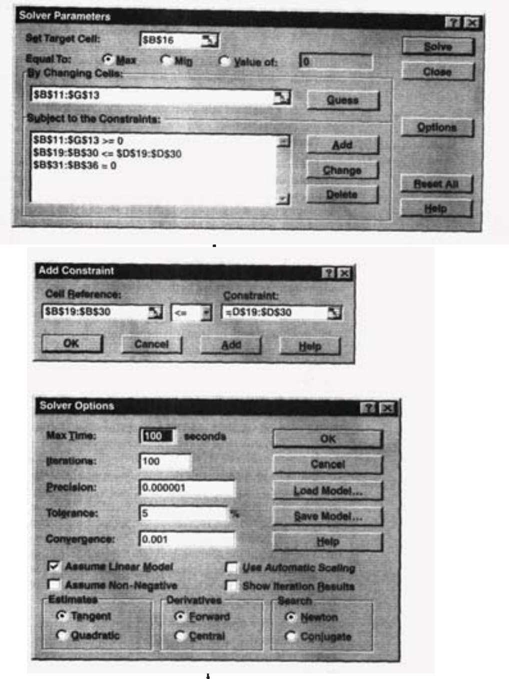
Note that the actual constraint is not shown explicitly in the spreadsheet; it is entered only in the **Solver** menu. However, the right-hand side of the constraint can be another cell in the spreadsheet or a constant. By specifying a range of cells for the right-hand side and a constant for the left-hand side, we can add a whole set of constraints in a single command. For instance, the range B11:G13 represents all the decision variables, so if we use this range as the left-hand side, a ≥ symbol, and a zero for the right-hand side, we will represent all the nonnegativity constraints (16.25) to (16.27). By choosing the **Add** button after each constraint we enter, we can add all the model constraints. When we are done, we choose the **OK** button, which returns us to the original form. We have the option to edit or delete constraints at any time.

Finally, before running the model, we must tell Excel that we want it to use the LP solution algorithm.[2] We do this by choosing the **Options** button to bring up another dialog box (see Figure 16.4) and choosing the **Assume Linear Model** option. This form also allows us to limit the time the model will run and to specify certain tolerances. If the model does not

[2] Excel can also solve nonlinear optimization problems and will apply the nonlinear algorithm as a default. Since LP is *much* more efficient, we definitely want to choose it as long as our model meets the requirements. All the formulations in this chapter are linear and therefore can use LP.

FIGURE 16.2

Specification of objectives and constraints in Excel



Solver Parameters

Set Target Cell: $B$16

Equal To: ● Max ○ Min ○ Value of: 0

By Changing Cells:

$B$11:$G$13

Subject to the Constraints:

$B$11:$G$13 >= 0
$B$19:$B$30 <= $D$19:$D$30
$B$31:$B$36 = 0

Add Constraint

Cell Reference: $B$19:$B$30 <= Constraint: =D$19:$D$30

Solver Options

Max Time: 100 seconds

Iterations: 100

Precision: 0.000001

Tolerance: 5 %

Convergence: 0.001

☑ Assume Linear Model
□ Use Automatic Scaling
□ Assume Non-Negative
□ Show Iteration Results

Estimates: ● Tangent ○ Quadratic

Derivatives: ● Forward ○ Central

Search: ● Newton ○ Conjugate

converge to an answer, the most likely reason is an error in one of the constraints. However, sometimes increasing the search time or reducing tolerances will fix the problem when the solver cannot find a solution. The reader should consult the Excel manual for more detailed documentation on this and other features, as well as information on upgrades that may have occurred since this writing. Choosing the **OK** button returns us to the original form.

Once we have done all this, we are ready to run the model by choosing the **Solve** button. The program will pause to set up the problem in the proper format and then will go through a sequence of trial solutions (although not for long in such a small problem as this).

Basically, LP works by first finding a feasible solution—one that satisfies all the constraints—and then generating a succession of new solutions, each better than the last. When no further improvement is possible, it stops and the solution is optimal: It maximizes or minimizes the objective function. Appendix 16A provides background on how this process works.

The algorithm will stop with one of three answers:

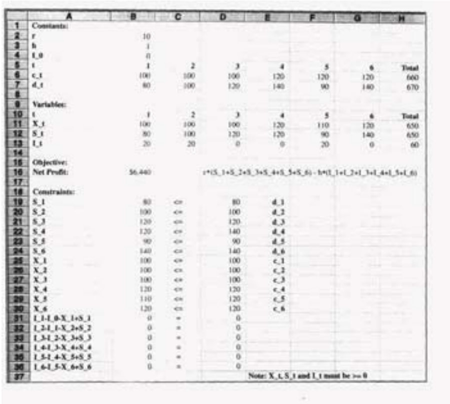
1. *Could not find a feasible solution.* This probably means that the problem is infeasible; that is, there is no solution that satisfies all the constraints. This could be due to a typing error (e.g., a plus sign was incorrectly typed as a minus sign) or a real infeasibility (e.g., it is not possible to meet demand with capacity). Notice that by clever formulation, one can avoid having the algorithm terminate with this depressing message when real infeasibilities exist. For instance, in formulation (16.6) to (16.27), we did not force sales to be equal to demand. Since cumulative demand exceeds cumulative capacity, it is obvious that this would not have been feasible. By setting separate sales and production variables, we let the computer tell us where demand cannot be met. Many variations on this trick are possible.

2. *Does not converge.* This means either that the algorithm could not find an optimal solution within the allotted time (so increasing the time or decreasing the tolerances under the **Options** menu might help) or that the algorithm is able to continue finding better and better solutions indefinitely. This second possibility can occur when the problem is **unbounded:** The objective can be driven to infinity by letting some variables grow positive or negative without bound. Usually this is the result of a failure to properly constrain a decision variable. For instance, in the above model, if we forgot to specify that all decision variables must be nonnegative, then the model will be able to make the objective arbitrarily large by choosing negative values of $I_t, t = 1, \ldots, 6$. Of course, we do not generate revenue via negative inventory levels, so it is important that nonnegativity constraints be included to rule out this nonsensical behavior.[3]

3. *Found a solution.* This is the outcome we want. When it occurs, the program will write the optimal values of the decision variables, objective value, and constraints into the spreadsheet. Figure 16.5 shows the spreadsheet as modified by the LP algorithm. The program also offers three reports—Answer, Sensitivity, and Limits—which write information about the solution into other spreadsheets. For instance, highlighting the Answer report generates a spreadsheet with the information shown in Figures 16.6 and 16.7. Figure 16.8 contains some of the information contained in the report generated by choosing Sensitivity.

Now that we have generated a solution, let us interpret it. Both Figure 16.5—the final spreadsheet—and Figure 16.6 show the optimal decision variables. From these we see that it is not optimal to produce at full capacity in every period. Specifically, the solution calls for producing only 110 units in month 5 when capacity is 120. This might seem odd given that demand exceeds capacity. However, if we look more carefully, we see that cumulative demand for periods 1 to 4 is 440 units, while cumulative capacity

[3] We will show how to modify the formulation to allow for backordering, which is like allowing negative inventory positions, without this inappropriately affecting the objective function, later in this chapter.



	A	B	C	D	E	F	G	H
1	Constants:							
2	r	10						
3	h	1						
4	I_0	0						
5	t	1	2	3	4	5	6	Total
6	c_t	100	100	100	120	120	120	660
7	d_t	80	100	120	140	90	140	670
8								
9	Variables:							
10	t	1	2	3	4	5	6	Total
11	X_t	100	100	100	120	110	120	650
12	S_t	80	100	120	120	90	140	650
13	I_t	20	20	0	0	20	0	60
14								
15	Objective:							
16	Net Profit:	$6,440		r*(S_1+S_2+S_3+S_4+S_5+S_6) - h*(I_1+I_2+I_3+I_4+I_5+I_6)				
17								
18	Constraints:							
19	S_1	80	<=	80	d_1			
20	S_2	100	<=	100	d_2			
21	S_3	120	<=	120	d_3			
22	S_4	120	<=	140	d_4			
23	S_5	90	<=	90	d_5			
24	S_6	140	<=	140	d_6			
25	X_1	100	<=	100	c_1			
26	X_2	100	<=	100	c_2			
27	X_3	100	<=	100	c_3			
28	X_4	120	<=	120	c_4			
29	X_5	110	<=	120	c_5			
30	X_6	120	<=	120	c_6			
31	I_1-I_0-X_1+S_1	0	=	0				
32	I_2-I_1-X_2+S_2	0	=	0				
33	I_3-I_2-X_3+S_3	0	=	0				
34	I_4-I_3-X_4+S_4	0	=	0				
35	I_5-I_4-X_5+S_5	0	=	0				
36	I_6-I_5-X_6+S_6	0	=	0				
37					Note: X_t, S_t and I_t must be >= 0			

for those periods is only 420 units. Thus, even when we run flat out for the first four months, we will fall short of meeting demand by 20 units. Demand in the final two months is only 230 units, while capacity is 240 units. Since our model does not permit backordering, it does not make sense to produce more than 230 units in months 5 and 6. Any extra units cannot be used to make up a previous shortfall.

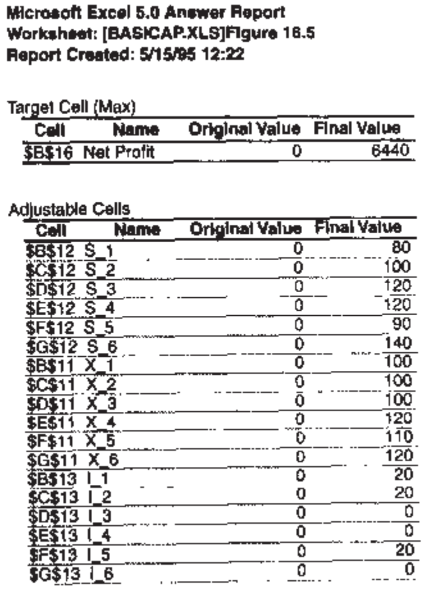
Figure 16.7 gives more details on the constraints by showing which ones are **binding** or **tight** (i.e., equal to the right-hand side) and which ones are **nonbinding** or **slack**, and by how much. Most interesting are the constraints on sales, given in (16.7) to (16.12), and capacity, in (16.13) to (16.18). As we have already noted, the capacity constraint on X_5 is nonbinding. Since we only produce 110 units in month 5 and have capacity for 120, this constraint is slack by 10 units. This means that if we changed this constraint by a little (e.g., reduced capacity in month 5 from 120 to 119 units), it would not change the optimal solution at all.

In this same vein, all sales constraints are tight except that for S_4. Since sales are limited to 140, but optimal sales are 120, this constraint has slackness of 20 units. Again, if we were to change this sales constraint by a little (e.g., limit sales to 141 units), the optimal solution would remain the same.

In contrast with these slack constraints, consider a binding constraint. For instance, consider the capacity constraint on X_1, which is the seventh one shown in Figure 16.7. Since the model chooses production equal to capacity in month 1, this constraint is tight. If we were to change this constraint by increasing or decreasing capacity, the solution would change. If we **relax** the constraint by increasing capacity, say, to 101 units, then we will be able to satisfy an additional unit of demand and therefore the net profit will

FIGURE 16.6

Optimal values report for LP example



Microsoft Excel 5.0 Answer Report
Worksheet: [BASICAP.XLS]Figure 16.5
Report Created: 5/15/95 12:22

Target Cell (Max)

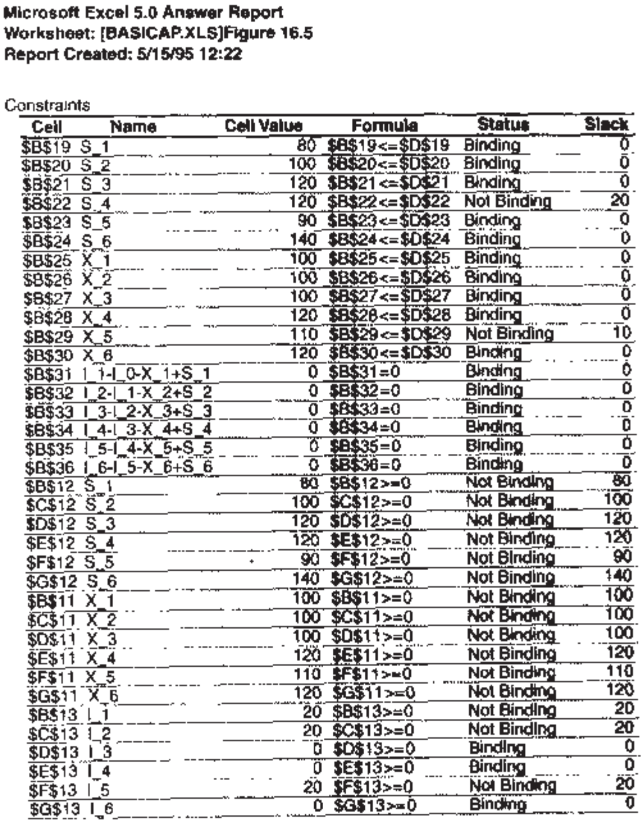
Cell	Name	Original Value	Final Value
$B$16	Net Profit	0	6440

Adjustable Cells

Cell	Name	Original Value	Final Value
$B$12	S_1	0	80
$C$12	S_2	0	100
$D$12	S_3	0	120
$E$12	S_4	0	120
$F$12	S_5	0	90
$G$12	S_6	0	140
$B$11	X_1	0	100
$C$11	X_2	0	100
$D$11	X_3	0	100
$E$11	X_4	0	120
$F$11	X_5	0	110
$G$11	X_6	0	120
$B$13	I_1	0	20
$C$13	I_2	0	20
$D$13	I_3	0	0
$E$13	I_4	0	0
$F$13	I_5	0	20
$G$13	I_6	0	0

FIGURE 16.7

Optimal constraint status for LP example



Microsoft Excel 5.0 Answer Report
Worksheet: [BASICAP.XLS]Figure 16.5
Report Created: 5/15/95 12:22

Constraints

Cell	Name	Cell Value	Formula	Status	Slack
$B$19	S_1	80	$B$19<=$D$19	Binding	0
$B$20	S_2	100	$B$20<=$D$20	Binding	0
$B$21	S_3	120	$B$21<=$D$21	Binding	0
$B$22	S_4	120	$B$22<=$D$22	Not Binding	20
$B$23	S_5	90	$B$23<=$D$23	Binding	0
$B$24	S_6	140	$B$24<=$D$24	Binding	0
$B$25	X_1	100	$B$25<=$D$25	Binding	0
$B$26	X_2	100	$B$26<=$D$26	Binding	0
$B$27	X_3	100	$B$27<=$D$27	Binding	0
$B$28	X_4	120	$B$28<=$D$28	Binding	0
$B$29	X_5	110	$B$29<=$D$29	Not Binding	10
$B$30	X_6	120	$B$30<=$D$30	Binding	0
$B$31	I_1-I_0-X_1+S_1	0	$B$31=0	Binding	0
$B$32	I_2-I_1-X_2+S_2	0	$B$32=0	Binding	0
$B$33	I_3-I_2-X_3+S_3	0	$B$33=0	Binding	0
$B$34	I_4-I_3-X_4+S_4	0	$B$34=0	Binding	0
$B$35	I_5-I_4-X_5+S_5	0	$B$35=0	Binding	0
$B$36	I_6-I_5-X_6+S_6	0	$B$36=0	Binding	0
$B$12	S_1	80	$B$12>=0	Not Binding	80
$C$12	S_2	100	$C$12>=0	Not Binding	100
$D$12	S_3	120	$D$12>=0	Not Binding	120
$E$12	S_4	120	$E$12>=0	Not Binding	120
$F$12	S_5	90	$F$12>=0	Not Binding	90
$G$12	S_6	140	$G$12>=0	Not Binding	140
$B$11	X_1	100	$B$11>=0	Not Binding	100
$C$11	X_2	100	$C$11>=0	Not Binding	100
$D$11	X_3	100	$D$11>=0	Not Binding	100
$E$11	X_4	120	$E$11>=0	Not Binding	120
$F$11	X_5	110	$F$11>=0	Not Binding	110
$G$11	X_6	120	$G$11>=0	Not Binding	120
$B$13	I_1	20	$B$13>=0	Not Binding	20
$C$13	I_2	20	$C$13>=0	Not Binding	20
$D$13	I_3	0	$D$13>=0	Binding	0
$E$13	I_4	0	$E$13>=0	Binding	0
$F$13	I_5	20	$F$13>=0	Not Binding	20
$G$13	I_6	0	$G$13>=0	Binding	0

increase. Since we will produce the extra item in month 1, hold it for three months to month 4 at a cost of $1 per month, and then sell it for $10, the overall increase in the objective from this change will be $10 − 3 = $7. Conversely, if we **tighten** the constraint by decreasing capacity, say to 99 units, then we will only be able to carry 19 units from month 1 to month 3 and will therefore lose one unit of demand in month 3. The loss in net profit from this unit will be $8 ($10 − $2 for two months' holding).

The sensitivity data generated by the LP algorithm shown in Figure 16.8 gives us more direct information on the sensitivity of the final solution to changes in the constraints. This report has a line for every constraint in the model and reports three important pieces of information:[4]

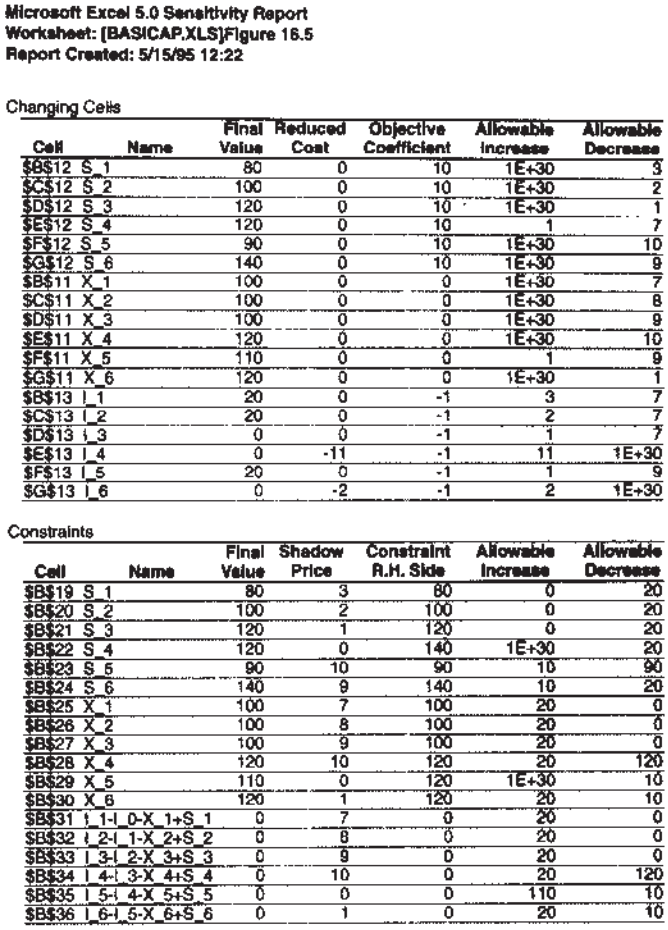
1. The **shadow price** represents the amount the optimal objective will be increased by a unit increase in the right-hand side of the constraint.
2. The **allowable increase** represents the amount by which the right-hand side can be increased before the shadow price no longer applies.
3. The **allowable decrease** represents the amount by which the right-hand side can be decreased before the shadow price no longer applies.

Appendix 16A gives a geometric explanation of how these numbers are computed.

[4] The report also contains sensitivity information about the coefficients in the objective function. See Appendix 16A for a discussion of this.

FIGURE 16.8

Sensitivity analysis for LP example



Microsoft Excel 5.0 Sensitivity Report
Worksheet: [BASICAP.XLS]Figure 16.5
Report Created: 5/15/95 12:22

Changing Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$B$12	S_1	80	0	10	1E+30	3
$C$12	S_2	100	0	10	1E+30	2
$D$12	S_3	120	0	10	1E+30	1
$E$12	S_4	120	0	10	1	7
$F$12	S_5	90	0	10	1E+30	10
$G$12	S_6	140	0	10	1E+30	9
$B$11	X_1	100	0	0	1E+30	7
$C$11	X_2	100	0	0	1E+30	8
$D$11	X_3	100	0	0	1E+30	9
$E$11	X_4	120	0	0	1E+30	10
$F$11	X_5	110	0	0	1	9
$G$11	X_6	120	0	0	1E+30	1
$B$13	I_1	20	0	-1	3	7
$C$13	I_2	20	0	-1	2	7
$D$13	I_3	0	0	-1	1	7
$E$13	I_4	0	-11	-1	11	1E+30
$F$13	I_5	20	0	-1	1	9
$G$13	I_6	0	-2	-1	2	1E+30

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$B$19	S_1	80	3	80	0	20
$B$20	S_2	100	2	100	0	20
$B$21	S_3	120	1	120	0	20
$B$22	S_4	120	0	140	1E+30	20
$B$23	S_5	90	10	90	10	90
$B$24	S_6	140	9	140	10	20
$B$25	X_1	100	7	100	20	0
$B$26	X_2	100	8	100	20	0
$B$27	X_3	100	9	100	20	0
$B$28	X_4	120	10	120	20	120
$B$29	X_5	110	0	120	1E+30	10
$B$30	X_6	120	1	120	20	10
$B$31	I_1-I_0-X_1+S_1	0	7	0	20	0
$B$32	I_2-I_1-X_2+S_2	0	8	0	20	0
$B$33	I_3-I_2-X_3+S_3	0	9	0	20	0
$B$34	I_4-I_3-X_4+S_4	0	10	0	20	120
$B$35	I_5-I_4-X_5+S_5	0	0	0	110	10
$B$36	I_6-I_5-X_6+S_6	0	1	0	20	10

To see how these data are interpreted, consider the information in Figure 16.8 on the seventh line of the constraint section for the capacity constraint $X_1 \leq 100$. The shadow price is \$7, which means that if the.constraint is changed to $X_1 \leq 101$, net profit will increase by \$7, precisely as we computed above. The allowable increase is 20 units, which means that each unit capacity increase in period 1 up to a total of 20 units increases net profit by \$7. Therefore, an increase in capacity from 100 to 120 will increase net profit by $20 \times 7 = \$140$. Above 20 units, we will have satisfied all the lost demand in month 4, and therefore further increases will not improve profit. Thus, this constraint will become nonbinding once the right-hand side exceeds 120. Notice that the allowable decrease is zero for this constraint. What this means is that the shadow price of \$7 is not valid for decreases in the right-hand side. As we computed above, the decrease in net profit from a unit decrease in the capacity in month 1 is \$8. In general, we can only determine the impact of changes outside the allowable increase or decrease range by actually changing the constraints and rerunning the LP solver.

The above examples are illustrative of the following general behavior of linear programming models:

1. Changing the right-hand sides of nonbinding constraints by a small amount does not affect the optimal solution. The shadow price of a nonbinding constraint is always zero.

2. Increasing the right-hand side of a binding constraint will increase the objective by an amount equal to the shadow price times the size of the increase, provided that the increase is smaller than the allowable increase.
3. Decreasing the right-hand side of a binding constraint will decrease the objective by an amount equal to the shadow price times the size of the decrease, provided that the decrease is smaller than the allowable decrease.
4. Changes in the right-hand sides beyond the allowable increase or decrease range have an indeterminate effect and must be evaluated by resolving the modified model.
5. All these sensitivity results apply to changes in *one right-hand side variable at a time*. If multiple changes are made, the effects are not necessarily additive. Generally, multiple-variable sensitivity analysis must be done by resolving the model under the multiple changes.

16.3 Product Mix Planning

Now that we have set up the basic framework for formulating and solving aggregate planning problems, we can examine some commonly encountered situations. The first realistic aggregate planning issue we will consider-is that of product mix planning. To do this, we need to extend the model of the previous section to consider multiple products explicitly. As mentioned previously, allowing multiple products raises the possibility of a "floating bottleneck." That is, if the different products require different amounts of processing time on the various workstations, then the workstation that is most heavily loaded during a period may well depend on the mix of products run during that period. If flexibility in the mix is possible, we can use the AP module to adjust the mix in accordance with available capacity. And if the mix is essentially fixed, we can use the AP module to identify bottlenecks.

16.3.1 Basic Model

We start with a direct extension of the previous single-product model in which demands are assumed fixed and the objective is to minimize the inventory carrying cost of meeting these demands. To do this, we introduce the following notation:

i = an index of product, $i = 1, \ldots, m$, so m represents total number of products
j = an index of workstation, $j = 1, \ldots, n$, so n represents total number of workstations
t = an index of period, $t = 1, \ldots, \bar{t}$, so $\bar{t}$ represents planning horizon
$\bar{d}_{it}$ = maximum demand for product i in period t
$\underline{d}_{it}$ = minimum sales[5] allows of product i in period t
a_{ij} = time required on workstation j to produce one unit of product i
c_{jt} = capacity of workstation j in period t in units consistent with those used to define a_{ij}
r_i = net profit from one unit of product i
h_i = cost[6] to hold one unit of product i for one period t

[5] This might represent firm commitments that we do not want the computer program to violate.

[6] It is common to set h_i equal to the raw materials cost of product i times a one-period interest rate to represent the opportunity cost of the money tied up in inventory; but it may make sense to use higher values to penalize inventory that causes long, uncompetitive cycle times.

X_{it} = amount of product i produced in period t
S_{it} = amount of product i sold in period t
I_{it} = inventory of product i at end of period t (I_{i0} is given as data)

Again, X_{it}, S_{it}, and I_{it} are decision variables, while the other symbols are constants representing input data. We can give a linear program formulation of the problem to maximize net profit minus inventory carrying cost subject to upper and lower bounds on sales and capacity constraints as

$$\text{Maximize} \quad \sum_{t=1}^{t} \sum_{i=1}^{m} r_i S_{it} - h_i I_{it} \qquad (16.28)$$

Subject to:

$$\underline{d}_{it} \leq S_{it} \leq \bar{d}_{it} \qquad \text{for all } i, t \qquad (16.29)$$

$$\sum_{i=1}^{m} a_{ij} X_{it} \leq c_{jt} \qquad \text{for all } j, t \qquad (16.30)$$

$$I_{it} = I_{it-1} + X_{it} - S_{it} \qquad \text{for all } i, t \qquad (16.31)$$

$$X_{it}, S_{it}, I_{it} \geq 0 \qquad \text{for all } i, t \qquad (16.32)$$

In comparison to the previous single-product model, we have adjusted constraints (16.29) to include lower, as well as upper, bounds on sales. For instance, the firm may have long-term contracts that obligate it to produce certain minimum amounts of certain products. Conversely, the market for some products may be limited. To maximize profit, the computer has incentive to set production so that all these constraints will be tight at their upper limits. However, this may not be possible due to capacity constraints (16.30). Notice that unlike in the previous formulation, we now have capacity constraints for each workstation in each period. By noting which of these constraints are tight, we can identify those resources that limit production. Constraints (16.31) are the multiproduct version of the balance equations, and constraints (16.32) are the usual nonnegativity constraints.

We can use LP (16.28)–(16.32) to obtain several pieces of information, including

1. **Demand feasibility.** We can determine whether a set of demands is capacity-feasible. If the constraint $S_{it} \leq \bar{d}_{it}$ is tight, then the upper bound on demand $\bar{d}_{it}$ is feasible. If not, then it is capacity-infeasible. If demands given by the lower bounds on demand $\underline{d}_{it}$ are capacity-infeasible, then the computer program will return a "could not find a feasible solution" message and the user must make changes (e.g., reduce demands or increase capacity) in order to get a solution.
2. **Bottleneck locations.** Constraints (16.30) restrict production on each workstation in each period. By noting which of these constraints are binding, we can determine which workstations limit capacity in which periods. A workstation that is consistently binding in many periods is a clear bottleneck and requires close management attention.
3. **Product mix.** If we are unable, for capacity reasons, to attain all the upper bounds on demand, then the computer will reduce sales below their maximum for some products. It will try to maximize revenue by producing those products with high net profit, but because of the capacity constraints, this is not a simple matter, as we will see in the following example.

16.3.2 A Simple Example

Let us consider a simple product mix example that shows why one needs a formal optimization method instead of a simpler ad hoc approach for these problems. We simplify matters by assuming a planning horizon of only one period. While this is certainly not a realistic assumption in general, in situations where we know in advance that we will never carry inventory from one period to the next, solving separate one-period problems for each period *will* yield the optimal solution. For example, if demands and cost coefficients are constant from period to period, then there is no incentive to build up inventory and therefore this will be the case.

Consider a situation in which a firm produces two products, which we will call products 1 and 2. Table 16.1 gives descriptive data for these two products. In addition to the direct raw material costs associated with each product, we assume a $5,000 per week fixed cost for labor and capital. Furthermore, there are 2,400 minutes (five days per week, eight hours per day) of time available on workstations A to D. We assume that all these data are identical from week to week. Therefore, there is no reason to build inventory in one week to sell in a subsequent week. (If we can meet maximum demand this week with this week's production, then the same thing is possible next week.) Thus, we can restrict our attention to a single week, and the only issue is the appropriate amount of each product to produce.

A Cost Approach. Let us begin by looking at this problem from a simple cost standpoint. Net profit per unit of product 1 sold is $45 ($90 − 45), while net profit per unit of product 2 sold is $60 ($100 − 40). This would seem to indicate that we should emphasize production of product 2. Ideally, we would like to produce 50 units of product 2 to meet maximum demand, but we must check the capacity of the four workstations to make sure this is possible. Since workstation B requires the most time to make a unit of product 2 (30 minutes) among the four workstations, this is the potential constraint. Producing 50 units of product 2 on workstation B will require

$$30 \text{ minutes per unit} \times 50 \text{ units} = 1{,}500 \text{ minutes}$$

This is less than the available 2,400 minutes on workstation B, so producing 50 units of product 2 is feasible.

Now we need to determine how many units of product 1 we can produce with the leftover capacity. The unused time on workstations A to D after subtracting the time to

TABLE 16.1 Input Data for Single-Period AP Example

Product	1	2
Selling price	$90	$100
Raw material cost	$45	$40
Maximum weekly sales	100	50
Minutes per unit on workstation A	15	10
Minutes per unit on workstation B	15	30
Minutes per unit on workstation C	15	5
Minutes per unit on workstation D	15	5

make 50 units of product 2 we compute as

$$\begin{aligned} 2{,}400 - 10(50) &= 1{,}900 \text{ minutes on workstation A} \\ 2{,}400 - 30(50) &= 900 \text{ minutes on workstation B} \\ 2{,}400 - 5(50) &= 2{,}150 \text{ minutes on workstation C} \\ 2{,}400 - 5(50) &= 2{,}150 \text{ minutes on workstation D} \end{aligned}$$

Since one unit of product 1 requires 15 minutes of time on each of the four workstations, we can compute the maximum possible production of product 1 at each workstation by dividing the unused time by 15. Since workstation B has the least remaining time, it is the potential bottleneck. The maximum production of product 1 on workstation B (after subtracting the time to produce 50 units of product 2) is

$$\frac{900}{15} = 60$$

Thus, even though we can sell 100 units of product 1, we only have capacity for 60.

The weekly profit from making 60 units of product 1 and 50 units of product 2 is

$$\$45 \times 60 + \$60 \times 50 - \$5{,}000 = \$700$$

Is this the best we can do?

A Bottleneck Approach. The preceding analysis is entirely premised on costs and considers capacity only as an afterthought. A better method might be to look at cost *and* capacity, by computing a ratio representing ***profit per minute of bottleneck time used*** for each product. This requires that we first identify the bottleneck, which we do by computing the minutes required on each workstation to satisfy maximum demand and seeing which machine is most overloaded.[7] This yields

$$\begin{aligned} 15(100) + 10(50) &= 2{,}000 \text{ minutes on workstation A} \\ 15(100) + 30(50) &= 3{,}000 \text{ minutes on workstation B} \\ 15(100) + 5(50) &= 1{,}750 \text{ minutes on workstation C} \\ 15(100) + 5(50) &= 1{,}750 \text{ minutes on workstation D} \end{aligned}$$

Only workstation B requires more than the available 2,400 minutes, so we designate it the bottleneck. Hence, we would like to make the most profitable use of our time on workstation B. To determine which of the two products does this, we compute the ratio of net profit to minutes on workstation B as

$$\frac{\$45}{15} = \$3 \text{ per minute spent processing product 1}$$

$$\frac{\$60}{30} = \$2 \text{ per minute spent processing product 2}$$

This calculation indicates the reverse of our previous cost analysis. Each minute spent processing product 1 on workstation B nets us \$3, as opposed to only \$2 per minute spent on product 2. Therefore, we should emphasize production of product 1, not product 2. If we produce 100 units of product 1 (the maximum amount allowed by the demand constraint), then since all workstations require 15 min per unit of one, the unused time on each workstation is

$$2{,}400 - 15(100) = 900 \text{ minutes}$$

[7] The alert reader should be suspicious at this point, since we know that the identity of the "bottleneck" can depend on the product mix in a multiproduct case.

Then since workstation B is the slowest operation for producing product 2, this is what limits the amount we can produce. Each unit of product 2 requires 30 minutes on B; thus, we can produce

$$\frac{900}{30} = 30$$

units of product 2. The net profit from producing 100 units of product 1 and 30 units of product 2 is

$$\$45 \times 100 + \$60 \times 30 - \$5{,}000 = \$1{,}300$$

This is clearly better than the $700 we got from using our original analysis and, it turns out, is the best we can do. But will this method always work?

A Linear Programming Approach. To answer the question of whether the previous "bottleneck ratio" method will always determine the optimal product mix, we consider a slightly modified version of the previous example, with data shown in Table 16.2. The only changes in these data relative to the previous example are that the processing time of product 2 on workstation B has been increased from 30 to 35 minutes and the processing times for products 1 and 2 on workstation D have been increased from 15 and 5 to 25 and 14, respectively.

To execute our ratio-based approach on this modified problem, we first check for the bottleneck by computing the minutes required on each workstation to meet maximum demand levels:

$$\begin{aligned}
15(100) + 10(50) &= 2{,}000 \text{ minutes on workstation A}\\
15(100) + 35(50) &= 3{,}250 \text{ minutes on workstation B}\\
15(100) + 5(50) &= 1{,}750 \text{ minutes on workstation C}\\
25(100) + 14(50) &= 3{,}200 \text{ minutes on workstation D}
\end{aligned}$$

Workstation B is still the most heavily loaded resource, but now workstation D also exceeds the available 2,400 minutes.

If we designate workstation B as the bottleneck, then the ratio of net profit to minute of time on the bottleneck is

$$\frac{\$45}{15} = \$3.00 \text{ per minute spent processing product 1}$$

$$\frac{\$60}{35} = \$1.71 \text{ per minute spent processing product 2}$$

Table 16.2 Input Data for Modified Single-Period AP Example

Product	1	2
Selling price	$90	$100
Raw material cost	$45	$40
Maximum weekly sales	100	50
Minutes per unit on workstation A	15	10
Minutes per unit on workstation B	15	35
Minutes per unit on workstation C	15	5
Minutes per unit on workstation D	25	14

which, as before, indicates that we should produce as much product 1 as possible. However, now it is workstation D that is slowest for product 1. The maximum amount that can be produced on D in 2,400 minutes is

$$\frac{2{,}400}{25} = 96$$

Since 96 units of product 1 use up all available time on workstation D, we cannot produce any product 2. The net profit from this mix, therefore, is

$$\$45 \times 96 - \$5{,}000 = -\$680$$

This doesn't look very good—we are losing money. Moreover, while we used workstation B as our bottleneck for the purpose of computing our ratios, it was workstation D that determined how much product we could produce. Therefore, perhaps we should have designated workstation D as our bottleneck. If we do this, the ratio of net profit to minute of time on the bottleneck is

$$\frac{\$45}{25} = \$1.80 \text{ per minute spent processing product 1}$$

$$\frac{\$60}{14} = \$4.29 \text{ per minute spent processing product 2}$$

This indicates that it is more profitable to emphasize production of product 2. Since workstation B is slowest for product 2, we check its capacity to see how much product 2 we can produce, and we find

$$\frac{2{,}400}{35} = 68.57$$

Since this is greater than maximum demand, we should produce the maximum amount of product 2, which is 50 units. Now we compute the unused time on each machine as

$$2{,}400 - 10(50) = 1{,}900 \text{ minutes on workstation A}$$
$$2{,}400 - 35(50) = 650 \text{ minutes on workstation B}$$
$$2{,}400 - 5(50) = 2{,}150 \text{ minutes on workstation C}$$
$$2{,}400 - 14(50) = 1{,}700 \text{ minutes on workstation D}$$

Dividing the unused time by the minutes required to produce one unit of product 1 on each workstation gives us the maximum production of product 1 on each to be

$$\frac{1{,}900}{15} = 126.67 \text{ units on workstation A}$$

$$\frac{650}{15} = 43.33 \text{ units on workstation B}$$

$$\frac{2{,}150}{15} = 143.33 \text{ units on workstation C}$$

$$\frac{1{,}700}{25} = 68 \text{ units on workstation D}$$

Thus, workstation B limits production of product 1 to 43 units, so total net profit for this solution is

$$\$45 \times 43 + \$60 \times 50 - \$5{,}000 = -\$65$$

This is better, but we are still losing money. Is this the best we can do?

Finally, let's bring out our big gun (not really that big, since it is included in popular spreadsheet programs) and solve the problem with a linear programming package.

Letting X_1 (X_2) represent the quantity of product 1 (2) produced, we formulate a linear programming model to maximize profit subject to the demand and capacity constraints as

$$\text{Maximize} \quad 45X_1 + 60X_2 - 5{,}000 \tag{16.33}$$

Subject to:

$$X_1 \leq 100 \tag{16.34}$$

$$X_2 \leq 50 \tag{16.35}$$

$$15X_1 + 10X_2 \leq 2{,}400 \tag{16.36}$$

$$15X_1 + 35X_2 \leq 2{,}400 \tag{16.37}$$

$$15X_1 + 5X_2 \leq 2{,}400 \tag{16.38}$$

$$25X_1 + 14X_2 \leq 2{,}400 \tag{16.39}$$

Problem (16.33)–16.39) is trivial for any LP package. Ours (Excel) reports the solution to this problem to be

$$\text{Optimal objective} = \$557.94$$
$$X_1^* = 75.79$$
$$X_2^* = 36.09$$

Even if we round this solution down (which will certainly still be capacity-feasible, since we are reducing production amounts) to integer values

$$X_1^* = 75$$
$$X_2^* = 36$$

we get an objective of

$$\$45 \times 75 + \$60 \times 36 - \$5{,}000 = \$535$$

So making as much product 1 as possible and making as much product 2 as possible both result in negative profit. But making a *mix* of the two products generates positive profit!

The moral of this exercise is that even simple product mix problems can be subtle. No trick that chooses a dominant product or identifies the bottleneck before knowing the product mix can find the optimal solution in general. While such tricks can work for specific problems, they can result in extremely bad solutions in others. The only method guaranteed to solve these problems optimally is an exact algorithm such as those used in linear programming packages. Given the speed, power, and user-friendliness of modern LP packages, one should have a very good reason to forsake LP for an approximate method.

16.3.3 Extensions to the Basic Model

A host of variations on the basic problem given in formulation (16.28)–(16.32) are possible. We discuss a few of these next; the reader is asked to think of others in the problems at chapter's end.

Other Resource Constraints. Formulation (16.28)–(16.32) contains capacity constraints for the workstations, but not for other resources, such as people, raw materials, and transport devices. In some systems, these may be important determinants of overall capacity and therefore should be included in the AP module.

Generically, if we let

b_{ij} = units of resource j required per unit of product i

k_{jt} = number of units of resource j available in period t

X_{it} = amount of product i produced in period t

we can express the capacity constraint on resource j in period t as

$$\sum_{i=1}^{m} b_{ij} X_{it} \leq k_{jt} \tag{16.40}$$

Notice that b_{ij} and k_{jt} are the nonworkstation analogs to a_{ij} and c_{jt} in formulation (16.28)–(16.32).

As a specific example, suppose an inspector must check products 1, 2, and 3, which require 1, 2, and 1.5 hours, respectively, per unit to inspect. If the inspector is available a total of 160 hours per month, then the constraint on this person's time in month t can be represented as

$$X_{1t} + 2X_{2t} + 1.5X_{3t} \leq 160$$

If this constraint is binding in the optimal solution, it means that inspector time is a bottleneck and perhaps something should be reorganized to remove this bottleneck. (The plant could provide help for the inspector, simplify the inspection procedure to speed it up, or use quality-at-the-source inspections by the workstation operators to eliminate the need for the extra inspection step.)

As a second example, suppose a firm makes four different models of circuit board, all of which require one unit of a particular component. The component contains leading-edge technology and is in short supply. If k_t represents the total number of these components that can be made available in period t, then the constraint represented by component availability in each period t can be expressed as

$$X_{1t} + X_{2t} + X_{3t} + X_{4t} \leq k_t$$

Many other resource constraints can be represented in analogous fashion.

Utilization Matching. As our discussion so far shows, it is straightforward to model capacity constraints in LP formulations of AP problems. However, we must be careful about how we use these constraints in actual practice, for two reasons.

1. *Low-level complexity.* An AP module will necessarily gloss over details that can cause inefficiency in the short term. For instance, in the product mix example of the previous section, we assumed that it was possible to run the four machines 2,400 minutes per week. However, from our factory physics discussions of Part II, we know that it is virtually impossible to avoid some idle time on machines. Any source of randomness (machine failures, setups, errors in the scheduling process, etc.) can diminish utilization. While we cannot incorporate these directly in the AP model, we can account for their aggregate effect on utilization.
2. *Production control decisions.* As we noted in Chapter 13, it may be economically attractive to set the production quota below full average capacity, in order to achieve predictable customer service without excessive overtime costs. If the quota-setting module indicates that we should run at less than full utilization, we should include this fact in the aggregate planning module in order to maintain consistency.

These considerations may make it attractive to plan for production levels below full capacity. Although the decision of how close to capacity to run can be tricky, the mechanics of reducing capacity in the AP model are simple. If the c_{jt} parameters represent practical estimates of realistic full capacity of workstation j in period t, adjusted for setups, worker breaks, machine failures, and other reasonable detractors, then we can simply deflate capacity by multiplying these by a constant factor. For instance, if either historical experience or the quote-setting module indicates that it is reasonable to run at a fraction q of full capacity, then we can replace constraints (16.30) in LP (16.28)–(16.32) by

$$\sum_{i=1}^{m} a_{ij} X_{it} \leq q c_{jt} \qquad \text{for all } j, t$$

The result will be that a binding capacity constraint will occur whenever a workstation is loaded to $100q$ percent of capacity in a period.

Backorders. In LP (16.28)–(16.32), we forced inventory to remain positive at all times. Implicitly, we were assuming that demands had to be met from inventory or lost; no backlogging of unmet demand was allowed. However, in many realistic situations, demand is not lost when not met on time. Customers expect to receive their orders even if they are late. Moreover, it is important to remember that aggregate planning is a long-term planning function. Just because the model says a particular order will be late, that does not mean that this must be so in practice. If the model predicts that an order due nine months from now will be backlogged, there may be ample time to renegotiate the due date. For that matter, the demand may really be only a forecast, to which a firm customer due date has not yet been attached. With this in mind, it makes sense to think of the aggregate planning module as a tool for reconciling projected demands with available capacity. By using it to identify problems that are far in the future, we can address them while there is still time to do something about them.

We can easily modify LP (16.28)–(16.32) to permit backordering as follows:

$$\text{Maximize} \qquad \sum_{t=1}^{\bar{t}} r_i S_{it} - h_i I_{it}^{+} - \pi_{it}^{-} \tag{16.41}$$

Subject to:

$$\underline{d}_{it} \leq S_{it} \leq \bar{d}_{it} \qquad \text{for all } i, t \tag{16.42}$$

$$\sum_{i=1}^{m} a_{ij} X_{it} \leq c_{jt} \qquad \text{for all } j, t \tag{16.43}$$

$$I_{it} = I_{it-1} + X_{it} - S_{it} \qquad \text{for all } i, t \tag{16.44}$$

$$I_{it} = I_{it}^{+} - I_{it}^{-} \qquad \text{for all } i, t \tag{16.45}$$

$$X_{it}, S_{it}, I_{it}^{+}, I_{it}^{-} \geq 0 \qquad \text{for all } i, t \tag{16.46}$$

The main change was to redefine the inventory variable I_{it} as the difference $I_{it}^{+} - I_{it}^{-}$, where I_{it}^{+} represents the inventory of product i carried from period t to $t + 1$ and I_{it}^{-} represents the number of backorders carried from period t to $t + 1$. Both I_{it}^{+} and I_{it}^{-} must be nonnegative. However, I_{it} can be either positive or negative, and so we refer to it as the **inventory position** of product i in period t. A positive inventory position indicates on-hand inventory, while a negative inventory position indicates outstanding backorders. The coefficient π_i is the backorder analog to the holding cost h_i and represents the penalty to carry one unit of product i on backorder for one period of time. Because both I_{it}^{-}

and I_{it}^{+} appear in the objective with negative coefficients, the LP solver will never make both of them positive for the same period. This simply means that we won't both carry inventory and incur a backorder penalty in the same period.

In terms of modeling, the most troublesome parameters in this formulation are the backorder penalty coefficients π_i. What is the cost of being late by one period on one unit of product i? For that matter, why should the lateness penalty be linear in the number of periods late or the number of units that are late? Clearly, asking someone in the organization for these numbers is out of the question. Therefore, one should view this type of model as a tool for generating various long-term production plans. By increasing or decreasing the π_i coefficients relative to the h_i coefficients, the analyst can increase or decrease the relative penalty associated with backlogging. High π_i values tend to force the model to build up inventory to meet surges in demand, while low π_i values tend to allow the model to be late on satisfying some demands tl.at occur during peak periods. By generating both types of plans, the user can get an idea of what options are feasible and select among them.

To accomplish this, we need not get overly fine with the selection of cost coefficients. We could set them with the simple equations

$$h_i = \alpha p_i \tag{16.47}$$

$$\pi_i = \beta \tag{16.48}$$

where α represents the one-period interest rate, suitably inflated to penalize uncompetitive cycle times caused by excess inventory, and p_i represents the raw materials cost of one unit of product i, so that αp_i represents the interest lost on the money tied up by holding one unit of product i in inventory. Analogously, β represents a (somewhat artificial) cost per period of delay on any product. The assumption here is that the true cost of being late (expediting costs, lost customer goodwill, lost future orders, etc.) is independent of the cost or price of the product. If Equations (16.47) and (16.48) are valid, then the user can fix α and generate many different production plans by varying the single parameter β.

Overtime. The previous representations of capacity assume each workstation is available a fixed amount of time in each period. Of course, in many systems there is the possibility of increasing the time via the use of overtime. Although we will treat overtime in greater detail in our upcoming discussion of workforce planning, it makes sense to note quickly that it is a simple matter to represent the option of overtime in a product mix model, even when labor is not being considered explicitly.

To do this, let

l'_j = cost of one hour of overtime at workstation j; a cost parameter

O_{jt} = overtime at workstation j in period t in hours; a decision variable

We can modify LP (16.41)–(16.46) to allow overtime at each workstation as follows:

$$\text{Maximize} \quad \sum_{t=1}^{\bar{t}} \{r_i S_{it} - h_i I_{it}^{+} - \pi_i I_{it}^{-} - \sum_{j=1}^{n} l'_j O_{jt}\} \tag{16.49}$$

Subject to:

$$\underline{d}_{it} \le S_{it} \le \bar{d}_{it} \quad \text{for all } i, t \tag{16.50}$$

$$\sum_{i=1}^{m} a_{ij} X_{it} \le c_{jt} + O_{jt} \quad \text{for all } j, t \tag{16.51}$$

$$I_{it} = I_{it-1} + X_{it} - S_{it} \quad \text{for all } i, t \tag{16.52}$$

$$I_{it} = I_{it}^{+} - I_{it}^{-} \qquad \text{for all } i, t \qquad (16.53)$$
$$X_{it}, S_{it}, I_{it}^{+}, I_{it}^{-} O_{jt} \geq 0 \qquad \text{for all } i, j, t \qquad (16.54)$$

The two changes we have made to LP (16.41)–(16.46) were to

1. Subtract the cost of overtime at stations $1, \ldots, n$, which is $\sum_{t=1}^{\bar{t}} \sum_{j=1}^{n} l'_j O_{jt}$, from the objective function.
2. Add the hours of overtime scheduled at station j in period t, denoted by O_{jt}, to the capacity of this resource c_{jt} in constraints (16.51).

It is natural to include both backlogging and overtime in the same model, since these are both ways of addressing capacity problems. In LP (16.49)–(16.54), the computer has the option of being late in meeting demand (backlogging) or increasing capacity via overtime. The specific combination it chooses depends on the relative cost of backordering (π_i) and overtime (l'_j). By varying these cost coefficients, the user can generate a range of production plans.

Yield Loss. In systems where product is scrapped at various points in the line due to quality problems, we must release extra material into the system to compensate for these losses. The result is that workstations upstream from points of yield loss are more heavily utilized than if there were no yield loss (because they must produce the extra material that will ultimately be scrapped). Therefore, to assess accurately the feasibility of a particular demand profile relative to capacity, we must consider yield loss in the aggregate planning module in systems where scrap is an issue.

We illustrate the basic effect of yield loss in Figure 16.9. In this simple line, α, β, and γ represent the fraction of product that is lost to scrap at workstations A, B, and C, respectively. If we require d units of product to come out of station C, then, on average, we will have to release $d/(1-\gamma)$ units into station C. To get $d/(1-\gamma)$ units out of station B, we will have to release $d/[(1-\beta)(1-\gamma)]$ units into B on average. Finally, to get the needed $d/[(1-\beta)(1-\gamma)]$ out of B, we will have to release $d/[(1-\alpha)(1-\beta)(1-\gamma)]$ units into A.

We can generalize the specific example of Figure 16.9 by defining

y_{ij} = cumulative yield from station j onward (including station j) for product i

If we want to get d units of product i out of the end of the line on average, then we must release

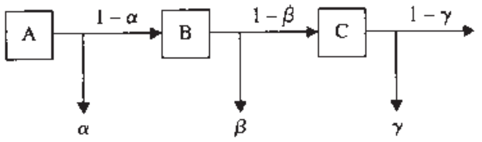
$$\frac{d}{y_{ij}} \qquad (16.55)$$

units of i into station j. These values can easily be computed in the manner used for the example in Figure 16.9 and updated in a spreadsheet or database as a function of the estimated yield loss at each station.

Using Equation (16.55) to adjust the production amounts X_{it} in the manner illustrated in Figure 16.9, we can modify the LP formulation (16.28)–(16.32) to consider

FIGURE 16.9

Yield loss in a three-station line



yield loss as follows:

$$\text{Maximize} \quad \sum_{t=1}^{\bar{t}} r_i S_{it} - h_i I_{it} \tag{16.56}$$

Subject to:

$$\underline{d}_{it} \leq S_{it} \leq \bar{d}_{it} \qquad \text{for all } i, t \tag{16.57}$$

$$\sum_{i=1}^{m} \frac{a_{ij} X_{it}}{y_{ij}} \leq c_{jt} \qquad \text{for all } j, t \tag{16.58}$$

$$I_{it} = I_{it-1} + X_{it} - S_{it} \qquad \text{for all } i, t \tag{16.59}$$

$$X_{it}, S_{it}, I_{it} \geq 0 \qquad \text{for all } i, t \tag{16.60}$$

As one would expect, the net effect of this change is to reduce the effective capacity of workstations, particularly those at the beginning of the line. By altering the y_{ij} values (or better yet, the individual yields that make up the y_{ij} values), the planner can get a feel for the sensitivity of the system to improvements in yields. Again as one would intuitively expect, the impact of reducing the scrap rate toward the end of the line is frequently much larger than that of reducing scrap toward the beginning of the line. Obviously, scrapping product late in the process is very costly and should be avoided wherever possible. If better process control and quality assurance in the front of the line can reduce scrap later, this is probably a sound policy. An aggregate planning module like that given in LP (16.56)–(16.60) is one way to get a sense of the economic and logistic impact of such a policy.

16.4 Workforce Planning

In systems where the workload is subject to variation, due to either a changing workforce size or overtime load, it may make sense to consider the aggregate planning (AP) and workforce planning (WP) modules in tandem. Questions of how and when to resize the labor pool or whether to use overtime instead of workforce additions can be posed in the context of a linear programming formulation to support both modules.

16.4.1 An LP Model

To illustrate how an LP model can help address the workforce-resizing and overtime allocation questions, we will consider a simple single-product model. In systems where product routings and processing times are either almost identical, so that products can be aggregated into a single product, or entirely separate, so that routings can be analyzed separately, the single-product model can be reasonable. In a system where bottleneck identification is complicated by different processing times and interconnected routings, a planner would most likely need an explicit multiproduct model. This involves a straightforward integration of a product mix model, like those we discussed earlier, with a workforce-planning model like that presented next.

We introduce the following notation, paralleling that which we have used up to now, with a few additions to address the workforce issues.

$j =$ an index of workstation, $j = 1, \ldots, n$, so n represents total number of workstations

$t =$ an index of period, $t = 1, \ldots, \bar{t}$, so $\bar{t}$ represents planning horizon

$\bar{d}_t =$ maximum demand in period t

$\underline{d}_t =$ minimum sales allowed in period t

$a_j =$ time required on workstation j to produce one unit of product

$b =$ number of worker-hours required to produce one unit of product

$c_{jt} =$ capacity of workstation j in period t

$r =$ net profit per unit of product sold

$h =$ cost to hold one unit of product for one period

$l =$ cost of regular time in dollars per worker-hour

$l' =$ cost of overtime in dollars per worker-hour

$e =$ cost to increase workforce by one worker-hour per period

$e' =$ cost to decrease workforce by one worker-hour per period

$X_t =$ amount produced in period t

$S_t =$ amount sold in period t

$I_t =$ inventory at end of t (I_0 is given as data)

$W_t =$ workforce in period t in worker-hours of regular time (W_0 is given as data)

$H_t =$ increase (hires) in workforce from period $t-1$ to t in worker-hours

$F_t =$ decrease (fires) in workforce from period $t-1$ to t in worker-hours

$O_t =$ overtime in period t in hours

We now have several new parameters and decision variables for representing the workforce considerations. First, we need b, the labor content of one unit of product, in order to relate workforce requirements to production needs. Once the model has used this parameter to determine the number of labor hours required in a given month, it has two options for meeting this requirement. Either it can schedule overtime, using the variable O_t and incurring cost at rate l'_t, or it can resize the workforce, using variables H_t and F_t and incurring a cost of e (e') for every worker added (laid off).

To model this planning problem as an LP, we will need to make the assumption that the cost of worker additions or deletions is linear in the number of workers added or deleted; that is, it costs twice as much to add (delete) two workers as it does to add (delete) one. Here we are assuming that e is an estimate of the hiring, training, outfitting, and lost productivity costs associated with bringing on a new worker. Similarly, e' represents the severance pay, unemployment costs, and so on associated with letting a worker go.

Of course, in reality, these workforce-related costs may not be linear. The training cost per worker may be less for a group than for an individual, since a single instructor can train many workers for roughly the same cost as a single one. On the other hand, the plant disruption and productivity falloff from introducing many new workers may be much more severe than those from introducing a single worker. Although one can use more sophisticated models to consider such sources of nonlinearity, we will stick with an LP model, keeping in mind that we are capturing general effects rather than elaborate details. Given that the AP and WP modules are used for long-term general planning purposes and rely on speculative forecasted data (e.g., of future demand), this is probably a reasonable choice for most applications.

We can write the LP formulation of the problem to maximize net profit, including labor, overtime, holding, and hiring/firing costs, subject to constraints on sales and capacity, as

$$\text{Maximize} \quad \sum_{t=1}^{\bar{t}} \{rS_t - hI_t - lW_t - l'O_t - eH_t - e'F_t\} \tag{16.61}$$

Subject to:

$$\underline{d}_t \le S_t \le \bar{d}_t \qquad \text{for all } t \tag{16.62}$$

$$a_j X_t \le c_{jt} \qquad \text{for all } j, t \tag{16.63}$$

$$I_t = I_{t-1} + X_t - S_t \qquad \text{for all } t \tag{16.64}$$

$$W_t = W_{t-1} + H_t - F_t \qquad \text{for all } t \tag{16.65}$$

$$bX_t \le W_t + O_t \qquad \text{for all } t \tag{16.66}$$

$$X_t, S_t, I_t, O_t, W_t, H_t, F_t \ge 0 \qquad \text{for all } t \tag{16.67}$$

The objective function in formulation (16.61) computes profit as the difference between net revenue and inventory carrying costs, wages (regular and overtime), and workforce increase/decrease costs. Constraints (16.62) are the usual bounds on sales. Constraints (16.63) are capacity constraints for each workstation. Constraints (16.64) are the usual inventory balance equations. Constraints (16.65) and (16.66) are new to this formulation. Constraints (16.65) define the variables W_t, $t = 1, \ldots, \bar{t}$, to represent the size of the workforce in period t in units of worker-hours. Constraints (16.66) constrain the worker-hours required to produce X_t, given by bX_t, to be less than or equal to the sum of regular time plus overtime, namely, $W_t + O_t$. Finally, constraints (16.67) ensure that production, sales, inventory, overtime, workforce size, and labor increases/decreases are all nonnegative. The fact that $I_t \ge 0$ implies no backlogging, but we could easily modify this model to account for backlogging in a manner like that used in LP (16.41)–(16.46).

16.4.2 A Combined AP/WP Example

To make LP (16.61)–(16.67) concrete and to give a flavor for the manner in which modeling, analysis, and decision making interact, we consider the example presented in the spreadsheet of Figure 16.10. This represents an AP problem for a single product with unit net revenue of \$1,000 over a 12-month planning horizon. We assume that each worker works 168 hours per month and that there are 15 workers in the system at the beginning of the planning horizon. Hence, the total number of labor hours available at the start of the problem is

$$W_0 = 15 \times 168 = 2{,}520$$

There is no inventory in the system at the start, so $I_0 = 0$.

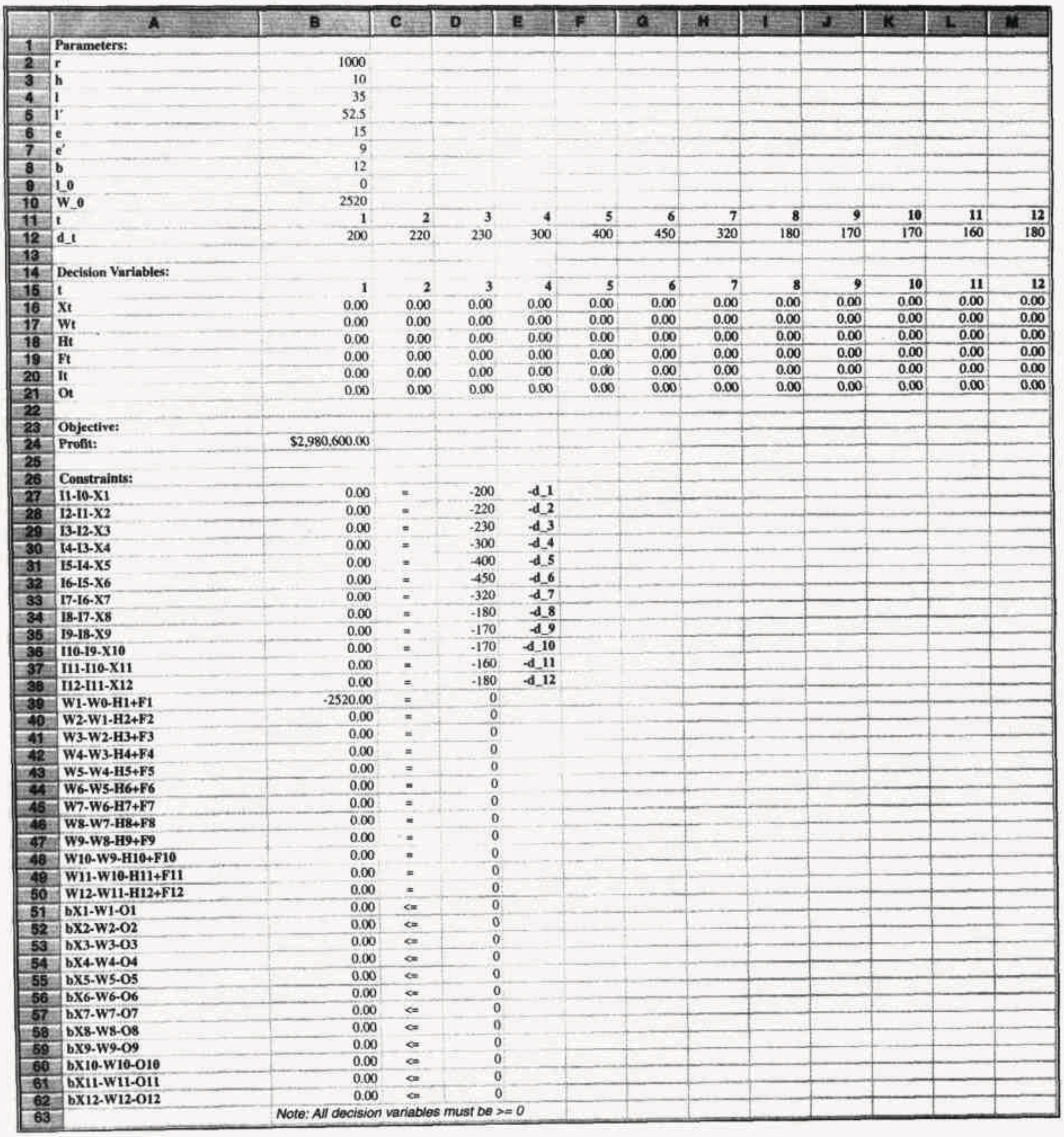
The cost parameters are estimated as follows. Monthly holding cost is \$10 per unit. Regular time labor (with benefits) costs \$35 per hour. Overtime is paid at time-and-a-half, which is equal to \$52.50 per hour. It costs roughly \$2,500 to hire and train a new worker. Since this worker will account for 168 hours per month, the cost in terms of dollars per worker-hour is

$$\frac{\$2{,}500}{168} = \$14.88 \approx \$15 \text{ per hour}$$

Since this number is only a rough approximation, we will round to an even \$15. Similarly, we estimate the cost to lay off a worker to be about \$1,500, so the cost per hour of

Figure 16.10

Initial spreadsheet for workforce planning example



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Parameters:**												
2	r	1000											
3	h	10											
4	l	35											
5	l′	52.5											
6	e	15											
7	e′	9											
8	b	12											
9	I_0	0											
10	W_0	2520											
11	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
12	d_t	200	220	230	300	400	450	320	180	170	170	160	180
13													
14	**Decision Variables:**												
15	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
16	Xt	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
17	Wt	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
18	Ht	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	Ft	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	It	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	Ot	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22													
23	**Objective:**												
24	**Profit:**	$2,980,600.00											
25													
26	**Constraints:**												
27	I1-I0-X1	0.00	=	-200	-d_1								
28	I2-I1-X2	0.00	=	-220	-d_2								
29	I3-I2-X3	0.00	=	-230	-d_3								
30	I4-I3-X4	0.00	=	-300	-d_4								
31	I5-I4-X5	0.00	=	-400	-d_5								
32	I6-I5-X6	0.00	=	-450	-d_6								
33	I7-I6-X7	0.00	=	-320	-d_7								
34	I8-I7-X8	0.00	=	-180	-d_8								
35	I9-I8-X9	0.00	=	-170	-d_9								
36	I10-I9-X10	0.00	=	-170	-d_10								
37	I11-I10-X11	0.00	=	-160	-d_11								
38	I12-I11-X12	0.00	=	-180	-d_12								
39	W1-W0-H1+F1	-2520.00	=	0									
40	W2-W1-H2+F2	0.00	=	0									
41	W3-W2-H3+F3	0.00	=	0									
42	W4-W3-H4+F4	0.00	=	0									
43	W5-W4-H5+F5	0.00	=	0									
44	W6-W5-H6+F6	0.00	=	0									
45	W7-W6-H7+F7	0.00	=	0									
46	W8-W7-H8+F8	0.00	=	0									
47	W9-W8-H9+F9	0.00	=	0									
48	W10-W9-H10+F10	0.00	=	0									
49	W11-W10-H11+F11	0.00	=	0									
50	W12-W11-H12+F12	0.00	=	0									
51	bX1-W1-O1	0.00	<=	0									
52	bX2-W2-O2	0.00	<=	0									
53	bX3-W3-O3	0.00	<=	0									
54	bX4-W4-O4	0.00	<=	0									
55	bX5-W5-O5	0.00	<=	0									
56	bX6-W6-O6	0.00	<=	0									
57	bX7-W7-O7	0.00	<=	0									
58	bX8-W8-O8	0.00	<=	0									
59	bX9-W9-O9	0.00	<=	0									
60	bX10-W10-O10	0.00	<=	0									
61	bX11-W11-O11	0.00	<=	0									
62	bX12-W12-O12	0.00	<=	0									
63		*Note: All decision variables must be >= 0*											

reduction in the monthly workforce is .

$$\frac{\$1{,}500}{168} = \$8.93 \approx \$9 \text{ per hour}$$

Again, we will use the rounded value of $9, since data are rough.

Notice that the projected demands (d_t) in the spreadsheet have a seasonal pattern to them, building to a peak in months 5 and 6, and tapering off thereafter. We will assume that backordering is not an option and that demands must be met, so the main issue will be how to do this.

Let us begin by expressing LP (16.61)–(16.67) in concrete terms for this problem. Because we are assuming that demands are met, we set $S_t = d_t$, which eliminates the need for separate sales variables S_t and sales constraints (16.62). Furthermore, to keep things simple, we will assume that the only capacity constraints are those posed by labor (i.e., it requires 12 hours of labor to produce each unit of product). No other machine or resource constraints need be considered. Thus we can omit constraints (16.63). Under these assumptions, the resulting LP formulation is

$$\begin{aligned}\text{Maximize}\quad & 1{,}000(d_1 + \cdots + d_{12}) - 10(I_1 + \cdots + I_{12})\\ & -35(W_1 + \cdots + W_{12}) - 52.5(O_1 + \cdots + O_{12})\\ & -15(H_1 + \cdots + H_{12}) - 9(F_1 + \cdots + F_{12})\end{aligned}\tag{16.68}$$

Subject to:

$$I_1 - I_0 - X_1 = -d_1 \tag{16.69}$$

$$I_2 - I_1 - X_2 = -d_2 \tag{16.70}$$

$$I_3 - I_2 - X_3 = -d_3 \tag{16.71}$$

$$I_4 - I_3 - X_4 = -d_4 \tag{16.72}$$

$$I_5 - I_4 - X_5 = -d_5 \tag{16.73}$$

$$I_6 - I_5 - X_6 = -d_6 \tag{16.74}$$

$$I_7 - I_6 - X_7 = -d_7 \tag{16.75}$$

$$I_8 - I_7 - X_8 = -d_8 \tag{16.76}$$

$$I_9 - I_8 - X_9 = -d_9 \tag{16.77}$$

$$I_{10} - I_9 - X_{10} = -d_{10} \tag{16.78}$$

$$I_{11} - I_{10} - X_{11} = -d_{11} \tag{16.79}$$

$$I_{12} - I_{11} - X_{12} = -d_{12} \tag{16.80}$$

$$W_1 - H_1 + F_1 = 2{,}520 \tag{16.81}$$

$$W_2 - W_1 - H_2 + F_2 = 0 \tag{16.82}$$

$$W_3 - W_2 - H_3 + F_3 = 0 \tag{16.83}$$

$$W_4 - W_3 - H_4 + F_4 = 0 \tag{16.84}$$

$$W_5 - W_4 - H_5 + F_5 = 0 \tag{16.85}$$

$$W_6 - W_5 - H_6 + F_6 = 0 \tag{16.86}$$

$$W_7 - W_6 - H_7 + F_7 = 0 \tag{16.87}$$

$$W_8 - W_7 - H_8 + F_8 = 0 \tag{16.88}$$

$$W_9 - W_8 - H_9 + F_9 = 0 \tag{16.89}$$

$$W_{10} - W_9 - H_{10} + F_{10} = 0 \tag{16.90}$$

$$W_{11} - W_{10} - H_{11} + F_{11} = 0 \tag{16.91}$$

$$W_{12} - W_{11} - H_{12} + F_{12} = 0 \tag{16.92}$$

$$12X_1 - W_1 - O_1 \leq 0 \tag{16.93}$$

$$12X_2 - W_2 - O_2 \leq 0 \tag{16.94}$$

$$12X_3 - W_3 - O_3 \leq 0 \tag{16.95}$$

$$12X_4 - W_4 - O_4 \leq 0 \tag{16.96}$$
$$12X_5 - W_5 - O_5 \leq 0 \tag{16.97}$$
$$12X_6 - W_6 - O_6 \leq 0 \tag{16.98}$$
$$12X_7 - W_7 - O_7 \leq 0 \tag{16.99}$$
$$12X_8 - W_8 - O_8 \leq 0 \tag{16.100}$$
$$12X_9 - W_9 - O_9 \leq 0 \tag{16.101}$$
$$12X_{10} - W_{10} - O_{10} \leq 0 \tag{16.102}$$
$$12X_{11} - W_{11} - O_{11} \leq 0 \tag{16.103}$$
$$12X_{12} - W_{12} - O_{12} \leq 0 \tag{16.104}$$
$$X_t, I_t, O_t, W_t, H_t, F_t \geq 0 \quad t = 1, \ldots, 12 \tag{16.105}$$

Objective (16.68) is identical to objective (16.61), except that the S_t variables have been replaced with d_t constants.[8] Constraints (16.69)–(16.80) are the usual balance constraints. For instance, constraint (16.69) simply states that

$$I_1 = I_0 + X_1 - d_1$$

That is, inventory at the end of month 1 equals inventory at the end of month 0 (i.e., the beginning of the problem) plus production during month 1, minus sales (demand) in month 1. We have arranged these constraints so that all decision variables are on the left-hand side of the equality and constants (d_t) are on the right-hand side. This is often a convenient modeling convention, as we will see in our analysis.

Constraints (16.81) to (16.92) are the labor balance equations given in constraints (16.65) of our general formulation. For instance, constraint (16.81) represents the relation

$$W_1 = W_0 + H_1 - F_1$$

so that the workforce at the end of month 1 (in units of worker-hours) is equal to the workforce at the end of month 0, plus any additions in month 1, minus any subtractions in month 1.

Constraints (16.93) to (16.104) ensure that the labor content of the production plan does not exceed available labor, which can include overtime. For instance, constraint (16.93) can be written as

$$12X_1 \leq W_1 + O_1$$

In the spreadsheet shown in Figure 16.10, we have entered the decision variables X_t, W_t, H_t, F_t, I_t, and O_t into cells B16:M21. Using these variables and the various coefficients from the top of the spreadsheet, we express objective (16.68) as a formula in cell B24. Notice that this formula reports a value equal to the unit profit times total demand, or

$$\begin{aligned}1{,}000(200 + 220 + 230 + 300 + 400 + 450 + 320\\ + 180 + 170 + 170 + 160 + 180) = \$2{,}980{,}000\end{aligned}$$

because all other terms in the objective are zero when the decision variables are set at zero.

We enter formulas for the left-hand sides of constraints (16.69) to (16.80) in cells B27:B38, the left-hand sides of constraints (16.81) to (16.92) in cells B39:B50, and the

[8] Since the d_t values are fixed, the first term in the objective function is not a function of our decision variables and could be left out without affecting the solution. We have kept it in so that our model reports a sensible profit function.

left-hand sides of constraints (16.93) to (16.104) in cells B51:B62. Notice that many of these constraints are not satisfied when all decision variables are equal to zero. This is hardly surprising, since we cannot expect to earn revenues from sales of product we have not made.

A convenient aspect of using a spreadsheet for solving LP models is that it provides us with a mechanism for playing with the model to gain insight into its behavior. For instance, in the spreadsheet of Figure 16.11 we try a **chase solution** where we set production equal to demand ($X_t = d_t$) and leave $W_t = W_0$ in every period. Although this satisfies the inventory balance constraints in cells B27:B38, and the workforce balance constraints in cells B39:B50, it violates the labor content constraints in cells B52:B57. The reason, of course, is that the current workforce is not sufficient to meet demand without using overtime. We could try adding overtime by adjusting the O_t variables in cells B21:M21. However, searching around for an optimal solution can be difficult, particularly in large models. Therefore, we will let the LP solver in the software do the work for us.

Using the procedure we described earlier, we specify constraints (16.69) to (16.105) in our model and turn it loose. The result is the spreadsheet in Figure 16.12. Based on the costs we chose, it turns out to be optimal not to use any overtime. (Overtime costs \$52.5 − 35 = 15.50 per hour each month, while hiring a new worker costs only \$15 per hour as a one-time cost.) Instead, the model adds 1,114.29 hours to the workforce, which represents

$$\frac{1,114.29}{168} = 6.6$$

new workers. After the peak season of months 4 to 7, the solution calls for a reduction of 1,474.29 + 120 = 1,594.29 hours, which implies laying off

$$\frac{1,594.29}{168} = 9.5$$

workers. Additionally, the solution involves building in excess of demand in months 1 to 4 and using this inventory to meet peak demand in months 5 to 7. The net profit resulting from this solution is \$1,687,337.14.

From a management standpoint, the planned layoffs in months 8 and 9 might be a problem. Although we have specified penalties for these layoffs, these penalties are highly speculative and may not accurately consider the long-term effects of hiring and firing on worker morale, productivity, and the firm's ability to recruit good people. Thus, it probably makes sense to carry our analysis further.

One approach we might consider would be to allow the model to hire but not fire workers. We can easily do this by eliminating the F_t variables or, since this requires fairly extensive changes in the spreadsheet, specifying additional constraints of the form

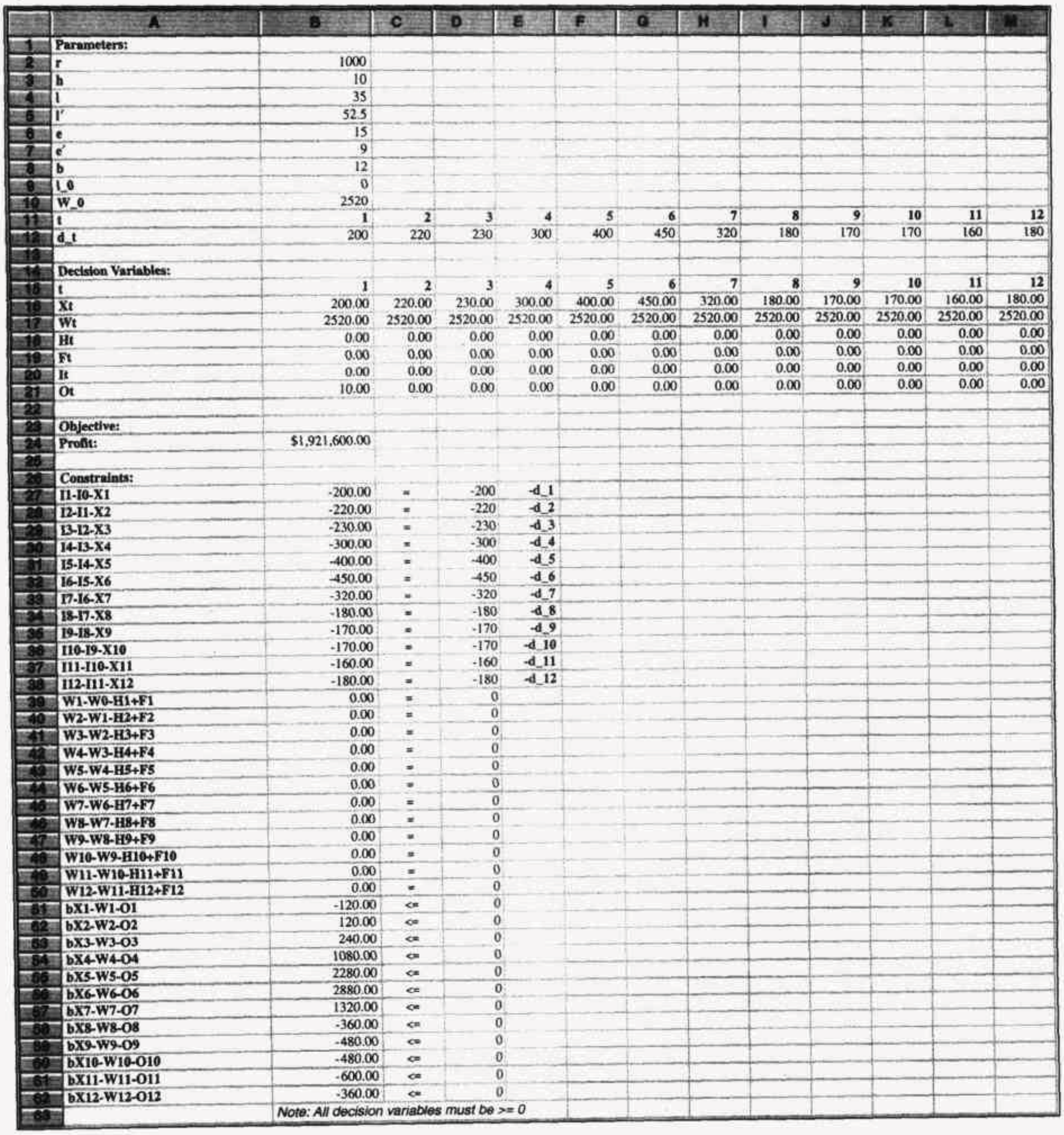
$$F_t = 0 \qquad t = 1, \ldots, 12$$

Rerunning the model with these additional constraints produces the spreadsheet in Figure 16.13. As we expect, this solution does not include any layoffs. Somewhat surprising, however, is the fact that it does not involve any new hires either (that is, $H_t = 0$ for every period). Instead of increasing the workforce size, the model has chosen to use overtime in months 3 to 7. Evidently, if we cannot fire workers, it is uneconomical to hire additional people.

However, when one looks more closely at the solution in Figure 16.13, a problem becomes evident. Overtime is too high. For instance, month 6 has more hours of overtime than hours of regular time! This means that our workforce of 15 people has

FIGURE 16.11

Infeasible "chase" solution



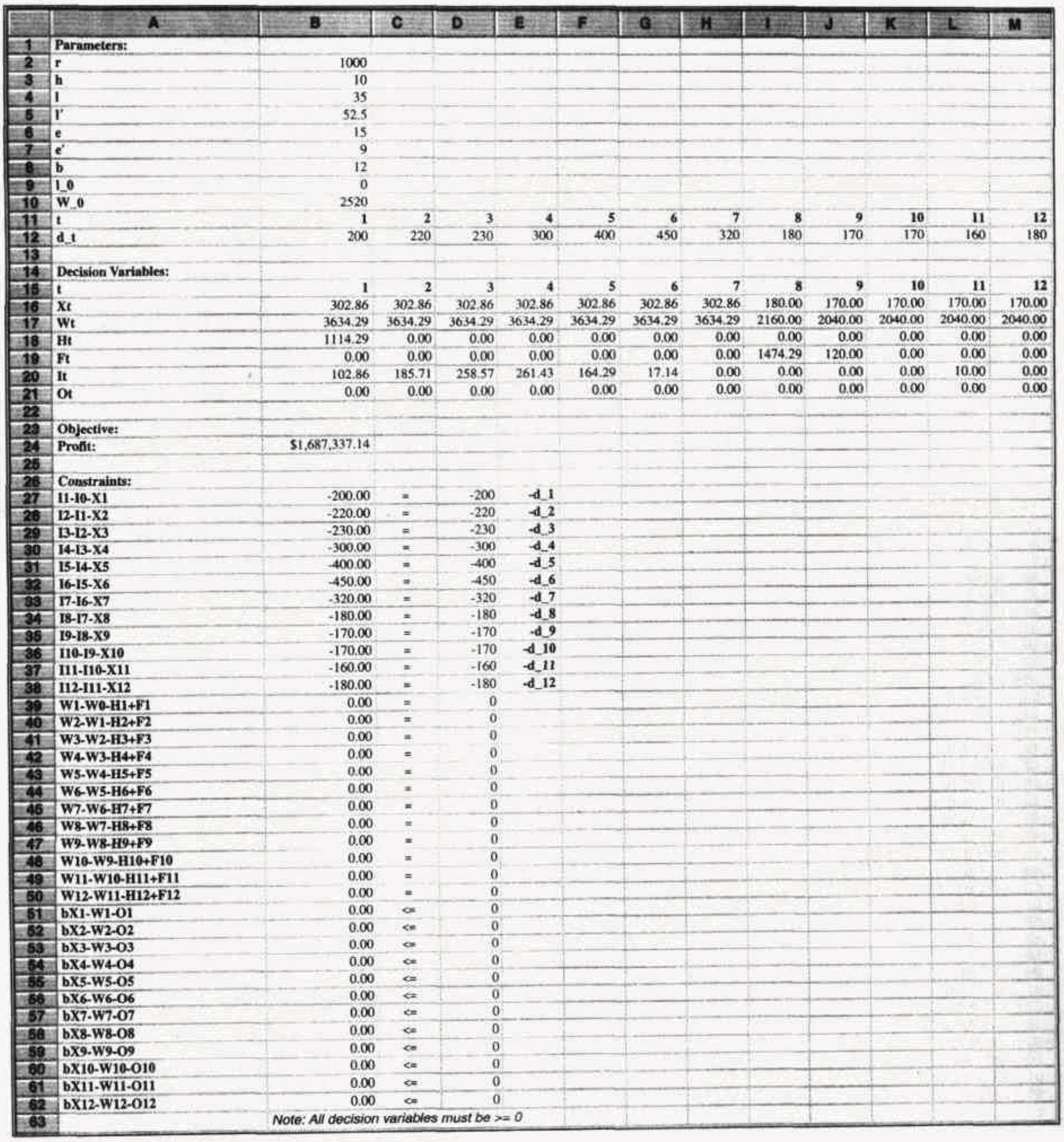
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Parameters:**												
2	r	1000											
3	h	10											
4	l	35											
5	l′	52.5											
6	e	15											
7	e′	9											
8	b	12											
9	I_0	0											
10	W_0	2520											
11	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
12	d_t	200	220	230	300	400	450	320	180	170	170	160	180
13													
14	**Decision Variables:**												
15	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
16	Xt	200.00	220.00	230.00	300.00	400.00	450.00	320.00	180.00	170.00	170.00	160.00	180.00
17	Wt	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00
18	Ht	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	Ft	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	It	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	Ot	10.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22													
23	**Objective:**												
24	**Profit:**	$1,921,600.00											
25													
26	**Constraints:**												
27	I1-I0-X1	-200.00	=	-200	-d_1								
28	I2-I1-X2	-220.00	=	-220	-d_2								
29	I3-I2-X3	-230.00	=	-230	-d_3								
30	I4-I3-X4	-300.00	=	-300	-d_4								
31	I5-I4-X5	-400.00	=	-400	-d_5								
32	I6-I5-X6	-450.00	=	-450	-d_6								
33	I7-I6-X7	-320.00	=	-320	-d_7								
34	I8-I7-X8	-180.00	=	-180	-d_8								
35	I9-I8-X9	-170.00	=	-170	-d_9								
36	I10-I9-X10	-170.00	=	-170	-d_10								
37	I11-I10-X11	-160.00	=	-160	-d_11								
38	I12-I11-X12	-180.00	=	-180	-d_12								
39	W1-W0-H1+F1	0.00	=	0									
40	W2-W1-H2+F2	0.00	=	0									
41	W3-W2-H3+F3	0.00	=	0									
42	W4-W3-H4+F4	0.00	=	0									
43	W5-W4-H5+F5	0.00	=	0									
44	W6-W5-H6+F6	0.00	=	0									
45	W7-W6-H7+F7	0.00	=	0									
46	W8-W7-H8+F8	0.00	=	0									
47	W9-W8-H9+F9	0.00	=	0									
48	W10-W9-H10+F10	0.00	=	0									
49	W11-W10-H11+F11	0.00	=	0									
50	W12-W11-H12+F12	0.00	=	0									
51	bX1-W1-O1	-120.00	<=	0									
52	bX2-W2-O2	120.00	<=	0									
53	bX3-W3-O3	240.00	<=	0									
54	bX4-W4-O4	1080.00	<=	0									
55	bX5-W5-O5	2280.00	<=	0									
56	bX6-W6-O6	2880.00	<=	0									
57	bX7-W7-O7	1320.00	<=	0									
58	bX8-W8-O8	-360.00	<=	0									
59	bX9-W9-O9	-480.00	<=	0									
60	bX10-W10-O10	-480.00	<=	0									
61	bX11-W11-O11	-600.00	<=	0									
62	bX12-W12-O12	-360.00	<=	0									
63		*Note: All decision variables must be >= 0*											

2,880/15 = 192 hours of overtime in the month, or about 48 hours per week per worker. This is obviously excessive.

One way to eliminate this overtime problem is to add some more constraints. For instance, we might specify that overtime is not to exceed 20 percent of regular time. This would correspond to the entire workforce working an average of one full day of

FIGURE 16.12

LP optimal solution



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Parameters:**												
2	r	1000											
3	h	10											
4	l	35											
5	l'	52.5											
6	e	15											
7	e'	9											
8	b	12											
9	I_0	0											
10	W_0	2520											
11	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
12	d_t	200	220	230	300	400	450	320	180	170	170	160	180
13													
14	**Decision Variables:**												
15	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
16	Xt	302.86	302.86	302.86	302.86	302.86	302.86	302.86	180.00	170.00	170.00	170.00	170.00
17	Wt	3634.29	3634.29	3634.29	3634.29	3634.29	3634.29	3634.29	2160.00	2040.00	2040.00	2040.00	2040.00
18	Ht	1114.29	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	Ft	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1474.29	120.00	0.00	0.00	0.00
20	It	102.86	185.71	258.57	261.43	164.29	17.14	0.00	0.00	0.00	0.00	10.00	0.00
21	Ot	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22													
23	**Objective:**												
24	**Profit:**	$1,687,337.14											
25													
26	**Constraints:**												
27	I1-I0-X1	-200.00	=	-200	-d_1								
28	I2-I1-X2	-220.00	=	-220	-d_2								
29	I3-I2-X3	-230.00	=	-230	-d_3								
30	I4-I3-X4	-300.00	=	-300	-d_4								
31	I5-I4-X5	-400.00	=	-400	-d_5								
32	I6-I5-X6	-450.00	=	-450	-d_6								
33	I7-I6-X7	-320.00	=	-320	-d_7								
34	I8-I7-X8	-180.00	=	-180	-d_8								
35	I9-I8-X9	-170.00	=	-170	-d_9								
36	I10-I9-X10	-170.00	=	-170	-d_10								
37	I11-I10-X11	-160.00	=	-160	-d_11								
38	I12-I11-X12	-180.00	=	-180	-d_12								
39	W1-W0-H1+F1	0.00	=	0									
40	W2-W1-H2+F2	0.00	=	0									
41	W3-W2-H3+F3	0.00	=	0									
42	W4-W3-H4+F4	0.00	=	0									
43	W5-W4-H5+F5	0.00	=	0									
44	W6-W5-H6+F6	0.00	=	0									
45	W7-W6-H7+F7	0.00	=	0									
46	W8-W7-H8+F8	0.00	=	0									
47	W9-W8-H9+F9	0.00	=	0									
48	W10-W9-H10+F10	0.00	=	0									
49	W11-W10-H11+F11	0.00	=	0									
50	W12-W11-H12+F12	0.00	=	0									
51	bX1-W1-O1	0.00	<=	0									
52	bX2-W2-O2	0.00	<=	0									
53	bX3-W3-O3	0.00	<=	0									
54	bX4-W4-O4	0.00	<=	0									
55	bX5-W5-O5	0.00	<=	0									
56	bX6-W6-O6	0.00	<=	0									
57	bX7-W7-O7	0.00	<=	0									
58	bX8-W8-O8	0.00	<=	0									
59	bX9-W9-O9	0.00	<=	0									
60	bX10-W10-O10	0.00	<=	0									
61	bX11-W11-O11	0.00	<=	0									
62	bX12-W12-O12	0.00	<=	0									
63		*Note: All decision variables must be >= 0*											

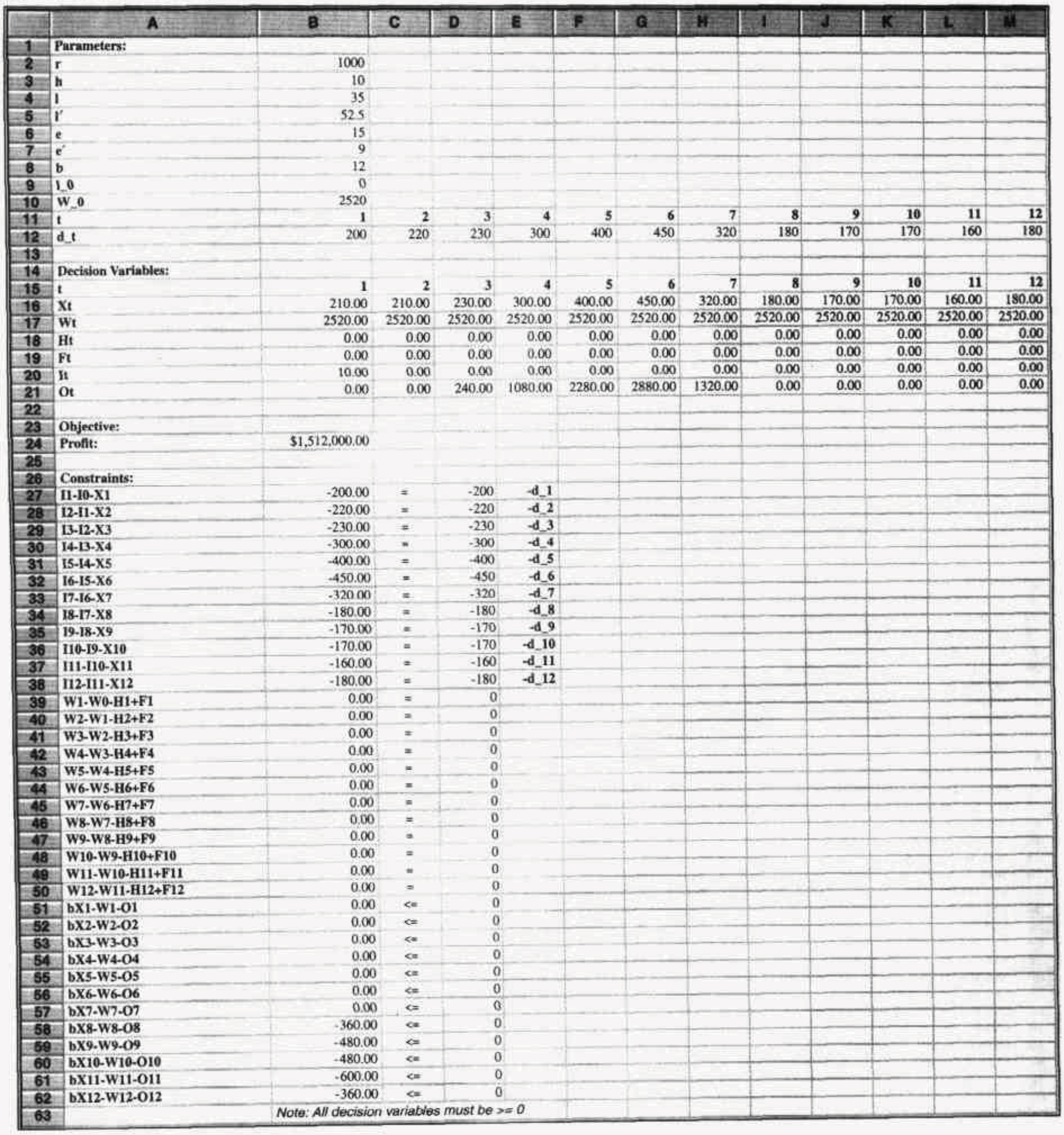
overtime per week in addition to the normal five-day workweek. We could do this by adding constraints of the form

$$O_t \leq 0.2W_t \qquad t = 1, \ldots, 12 \tag{16.106}$$

Doing this to the spreadsheet of Figure 16.13 and resolving results in the spreadsheet

FIGURE 16.13

Optimal solution when $F_t = 0$



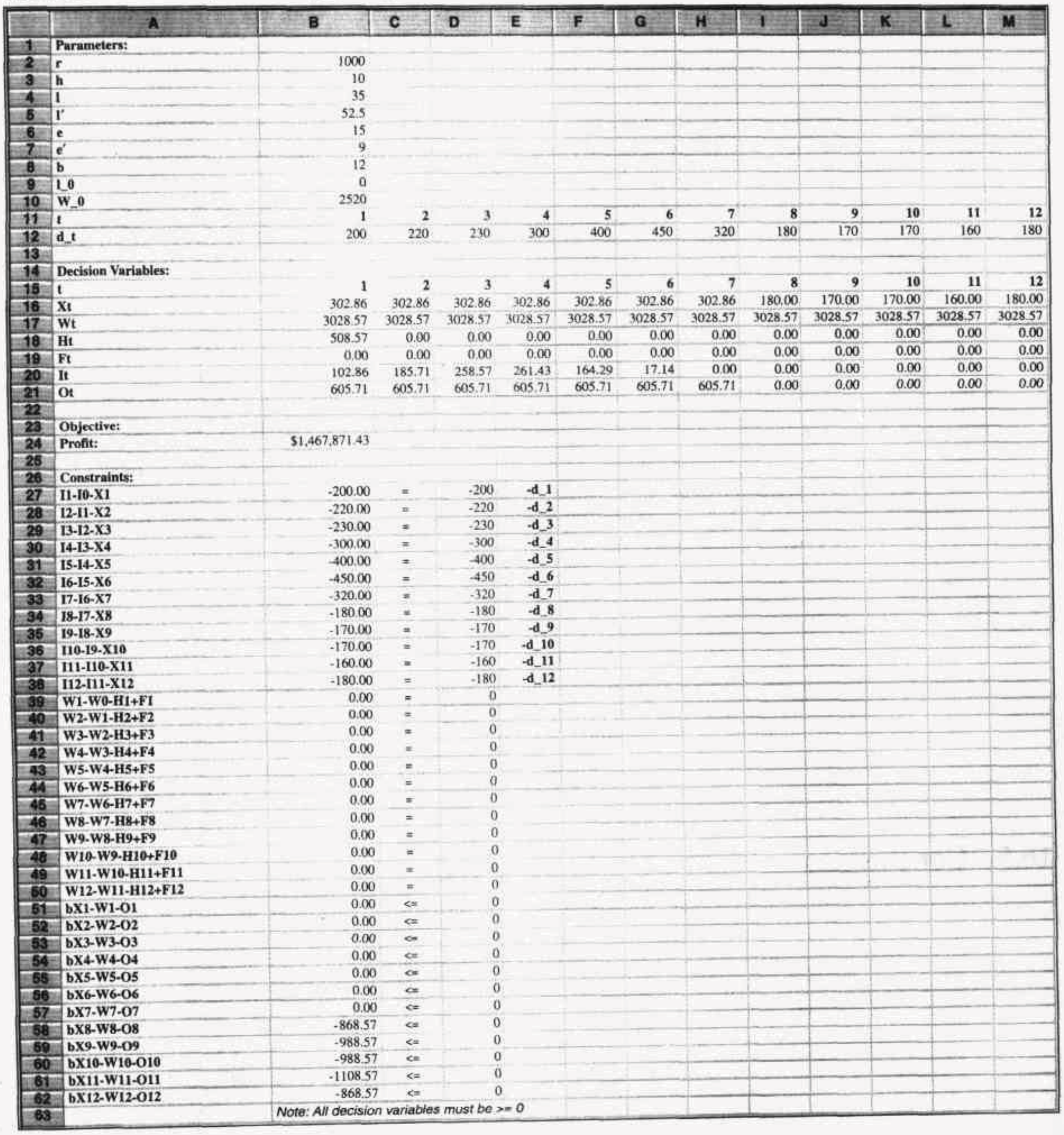
	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Parameters:**												
2	r	1000											
3	h	10											
4	l	35											
5	l′	52.5											
6	e	15											
7	e′	9											
8	b	12											
9	I_0	0											
10	W_0	2520											
11	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
12	d_t	200	220	230	300	400	450	320	180	170	170	160	180
13													
14	**Decision Variables:**												
15	t	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
16	Xt	210.00	210.00	230.00	300.00	400.00	450.00	320.00	180.00	170.00	170.00	160.00	180.00
17	Wt	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00	2520.00
18	Ht	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	Ft	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	It	10.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	Ot	0.00	0.00	240.00	1080.00	2280.00	2880.00	1320.00	0.00	0.00	0.00	0.00	0.00
22													
23	**Objective:**												
24	**Profit:**	$1,512,000.00											
25													
26	**Constraints:**												
27	I1-I0-X1	-200.00	=	-200	-d_1								
28	I2-I1-X2	-220.00	=	-220	-d_2								
29	I3-I2-X3	-230.00	=	-230	-d_3								
30	I4-I3-X4	-300.00	=	-300	-d_4								
31	I5-I4-X5	-400.00	=	-400	-d_5								
32	I6-I5-X6	-450.00	=	-450	-d_6								
33	I7-I6-X7	-320.00	=	-320	-d_7								
34	I8-I7-X8	-180.00	=	-180	-d_8								
35	I9-I8-X9	-170.00	=	-170	-d_9								
36	I10-I9-X10	-170.00	=	-170	-d_10								
37	I11-I10-X11	-160.00	=	-160	-d_11								
38	I12-I11-X12	-180.00	=	-180	-d_12								
39	W1-W0-H1+F1	0.00	=	0									
40	W2-W1-H2+F2	0.00	=	0									
41	W3-W2-H3+F3	0.00	=	0									
42	W4-W3-H4+F4	0.00	=	0									
43	W5-W4-H5+F5	0.00	=	0									
44	W6-W5-H6+F6	0.00	=	0									
45	W7-W6-H7+F7	0.00	=	0									
46	W8-W7-H8+F8	0.00	=	0									
47	W9-W8-H9+F9	0.00	=	0									
48	W10-W9-H10+F10	0.00	=	0									
49	W11-W10-H11+F11	0.00	=	0									
50	W12-W11-H12+F12	0.00	=	0									
51	bX1-W1-O1	0.00	<=	0									
52	bX2-W2-O2	0.00	<=	0									
53	bX3-W3-O3	0.00	<=	0									
54	bX4-W4-O4	0.00	<=	0									
55	bX5-W5-O5	0.00	<=	0									
56	bX6-W6-O6	0.00	<=	0									
57	bX7-W7-O7	0.00	<=	0									
58	bX8-W8-O8	-360.00	<=	0									
59	bX9-W9-O9	-480.00	<=	0									
60	bX10-W10-O10	-480.00	<=	0									
61	bX11-W11-O11	-600.00	<=	0									
62	bX12-W12-O12	-360.00	<=	0									
63		*Note: All decision variables must be >= 0*											

shown in Figure 16.14. The overtime limits have forced the model to resort to hiring. Since layoffs are still not allowed, the model hires only 508.57 hours worth of workers, or

$$\frac{508.57}{168} = 3$$

FIGURE 16.14

Optimal solution when $F_t = 0$ and $O_t \leq 0.2W_t$



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Parameters:**												
2	r	1000											
3	h	10											
4	l	35											
5	l'	52.5											
6	e	15											
7	e'	9											
8	b	12											
9	I_0	0											
10	W_0	2520											
11	t	1	2	3	4	5	6	7	8	9	10	11	12
12	d_t	200	220	230	300	400	450	320	180	170	170	160	180
13													
14	**Decision Variables:**												
15	t	1	2	3	4	5	6	7	8	9	10	11	12
16	Xt	302.86	302.86	302.86	302.86	302.86	302.86	302.86	180.00	170.00	170.00	160.00	180.00
17	Wt	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57	3028.57
18	Ht	508.57	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	Ft	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	It	102.86	185.71	258.57	261.43	164.29	17.14	0.00	0.00	0.00	0.00	0.00	0.00
21	Ot	605.71	605.71	605.71	605.71	605.71	605.71	605.71	0.00	0.00	0.00	0.00	0.00
22													
23	**Objective:**												
24	**Profit:**	$1,467,871.43											
25													
26	**Constraints:**												
27	I1-I0-X1	-200.00	=	-200	-d_1								
28	I2-I1-X2	-220.00	=	-220	-d_2								
29	I3-I2-X3	-230.00	=	-230	-d_3								
30	I4-I3-X4	-300.00	=	-300	-d_4								
31	I5-I4-X5	-400.00	=	-400	-d_5								
32	I6-I5-X6	-450.00	=	-450	-d_6								
33	I7-I6-X7	-320.00	=	-320	-d_7								
34	I8-I7-X8	-180.00	=	-180	-d_8								
35	I9-I8-X9	-170.00	=	-170	-d_9								
36	I10-I9-X10	-170.00	=	-170	-d_10								
37	I11-I10-X11	-160.00	=	-160	-d_11								
38	I12-I11-X12	-180.00	=	-180	-d_12								
39	W1-W0-H1+F1	0.00	=	0									
40	W2-W1-H2+F2	0.00	=	0									
41	W3-W2-H3+F3	0.00	=	0									
42	W4-W3-H4+F4	0.00	=	0									
43	W5-W4-H5+F5	0.00	=	0									
44	W6-W5-H6+F6	0.00	=	0									
45	W7-W6-H7+F7	0.00	=	0									
46	W8-W7-H8+F8	0.00	=	0									
47	W9-W8-H9+F9	0.00	=	0									
48	W10-W9-H10+F10	0.00	=	0									
49	W11-W10-H11+F11	0.00	=	0									
50	W12-W11-H12+F12	0.00	=	0									
51	bX1-W1-O1	0.00	<=	0									
52	bX2-W2-O2	0.00	<=	0									
53	bX3-W3-O3	0.00	<=	0									
54	bX4-W4-O4	0.00	<=	0									
55	bX5-W5-O5	0.00	<=	0									
56	bX6-W6-O6	0.00	<=	0									
57	bX7-W7-O7	0.00	<=	0									
58	bX8-W8-O8	-868.57	<=	0									
59	bX9-W9-O9	-988.57	<=	0									
60	bX10-W10-O10	-988.57	<=	0									
61	bX11-W11-O11	-1108.57	<=	0									
62	bX12-W12-O12	-868.57	<=	0									
63		*Note: All decision variables must be >= 0*											

new workers, as opposed to the 6.6 workers hired in the original solution in Figure 16.12. To attain the necessary production, the solution uses overtime in months 1 to 7. Notice that the amount of overtime used in these months is exactly 20 percent of regular time work hours, that is,

$$3{,}028.57 \times 0.2 = 605.71$$

What this means is that new constraints (16.106) are binding for periods 1 to 7, which we would be told explicitly if we printed out the sensitivity analysis reports generated by the LP solver. This implies that if it is possible to work more overtime in any of these months, we can improve the solution.

Notice that the net profit in the model of the spreadsheet shown in Figure 16.14 is \$1,467,871.43, which is a 13 percent decrease over the original optimal solution of \$1,687,337.14 in Figure 16.12. At first glance, it may appear that the policies of no layoffs and limits on overtime are expensive. On the other hand, it may really be telling us that our original estimates of the costs of hiring and firing were too low. If we were to increase these costs to represent, for example, long-term disruptions caused by labor changes, the optimal solution might be very much like the one arrived at in Figure 16.14.

16.4.3 Modeling Insights

In addition to providing a detailed example of a workforce formulation in LP (16.61)–(16.67), we hope that our discussion has helped the reader appreciate the following aspects of using an optimization model as the basis for an AP or WP module.

1. *Multiple modeling approaches.* There are often many ways to model a given problem, none of which is "correct" in any absolute sense. The key is to use cost coefficients and constraints to represent the main issues in a sensible way. In this example, we could have generated solutions without layoffs by either increasing the layoff penalty or placing constraints on the layoffs. Both approaches would achieve the same qualitative conclusions.

2. *Iterative model development.* Modeling and analysis almost never proceed in an ideal fashion in which the model is formulated, solved, and interpreted in a single pass. Often the solution from one version of the model suggests an alternate model. For instance, we had no way of knowing that eliminating layoffs would cause excessive overtime in the solution. We didn't know we would need constraints on the level of overtime until we saw the spreadsheet output of Figure 16.13.

16.5 Conclusions

In this chapter, we have given an overview of the issues involved in aggregate and workforce planning. A key observation behind our approach is that, because the aggregate planning and workforce planning modules use long time horizons, precise data and intricate modeling detail are impractical or impossible. We must recognize that the production or workforce plans that these modules generate will be adjusted as time evolves. The lower levels in the PPC hierarchy must handle the nuts-and-bolts challenge of converting the plans to action. The keys to a good AP module are to keep the focus on long-term planning (i.e., avoiding putting too many short-term control details in the model) and to provide links for consistency with other levels in the hierarchy. Some of the issues related to consistency were discussed in Chapter 13. Here, we close with some general observations about the aggregate and workforce planning functions:

1. *No single AP or WP module is right for every situation.* As the examples in this chapter show, aggregate and workforce planning can incorporate many different decision problems. A good AP or WP module is one that is tailored to address the specific issues faced by the firm.

2. *Simplicity promotes understanding.* Although it is desirable to address different issues in the AP/WP module. it is even more important to keep the model understandable. In general, these modules are used to generate candidate production and workforce plans, which will be examined, combined, and altered manually before being published as "The Plan." To generate a spectrum of plans (and explain them to others), the user must be able to trace changes in the model to changes in the plan. Because of this, it makes sense to start with as simple a formulation as possible. Additional detail (e.g., constraints) can be added later.

3. *Linear programming is a useful AP/WP tool.* The long planning horizon used for aggregate and workforce planning justifies ignoring many production details; therefore, capacity checks, sales restrictions, and inventory balances can be expressed as linear constraints. As long as we are willing to approximate actual costs with linear functions, an LP solver is a very efficient method for solving many problems related to the AP and WP modules. Because we are working with speculative long-range data, it generally does not make sense to use anything more sophisticated than LP (e.g., nonlinear or integer programming) in most aggregate and workforce planning situations.

4. *Robustness matters more than precision.* No matter how accurate the data and how sophisticated the model, the plan generated by the AP or WP module will never be followed exactly. The actual production sequence will be affected by unforeseen events that could not possibly have been factored into the module. This means that the mark of a good long-range production plan is that it enables us to do a reasonably good job even in the face of such contingencies. To find such a plan, the user of the AP module must be able to examine the consequences of various scenarios. This is another reason to keep the model reasonably simple.

Appendix 16A Linear Programming

Linear programming is a powerful mathematical tool for solving constrained optimization problems. The name derives from the fact that LP was first applied to find optimal schedules or "programs" of resource allocation. Hence, although LP generally does involve using a computer program, it does not entail programming on the part of the user in the sense of writing code.

In this appendix, we provide enough background to give the user of an LP package a basic idea of what the software is doing. Readers interested in more details should consult one of the many good texts on the subject (e.g., Eppen and Gould 1988 for an application-oriented overview, Murty 1983 for more technical coverage).

Formulation

The first step in using linear programming is to formulate a practical problem in mathematical terms. There are three basic choices we must make to do this:

1. **Decision variables** are quantities under our control. Typical examples for aggregate planning and workforce planning applications of LP are production quantities, number of workers to hire, and levels of inventory to hold.
2. **Objective function** is what we want to maximize or minimize. In most AP/WP applications, this is typically either to maximize profit or minimize cost. Beyond simply stating the objective, however, we must specify it in terms of the decision variables we have defined.

3. **Constraints** are restrictions on our choices of the decision variables. Typical examples for AP/WP applications include capacity constraints, raw materials limitations, restrictions on how fast we can add workers due to limitations on training capacity, and restrictions on physical flow (e.g., inventory levels as a direct result of how much we produce/procure and how much we sell).

When one is formulating an LP, it is often useful to try to specify the necessary inputs in the order in which they are listed. However, in realistic problems, one virtually never gets the "right" formulation in a single pass. The example in Section 16.4.2 illustrates some of the changes that may be required as a model evolves.

To describe the process of formulating an LP, let us consider the problem presented in Table 16.2. We begin by selecting decision variables. Since there are only two products and because demand and capacity are assumed stationary over time, the only decisions to make concern how much of each product to produce per week. Thus, we let X_1 and X_2 represent the weekly production quantities of products 1 and 2, respectively.

Next, we choose to maximize profit as our objective function. Since product 1 sells for $90 but costs $45 in raw material, its net profit is $45 per unit.[9] Similarly, product 2 sells for $100 but costs $40 in raw material, so its net unit profit is $60. Thus, weekly profit will be

$$45X_1 + 60X_2 - \text{weekly labor costs} - \text{weekly overhead costs}$$

But since we assume that labor and overhead costs are not affected by the choice of X_1 and X_2, we can use the following as our objective function for the LP model:

$$\text{Maximize} \quad 45X_1 + 60X_2$$

Finally, we need to specify constraints. If we could produce as much of products 1 and 2 as we wanted, we could drive the above objective function, and hence weekly profit, to infinity. This is not possible because of limitations on demand and capacity.

The demand constraints are easy. Since we can sell at most 100 units per week of product 1 and 50 units per week of product 2, our decision variables X_1 and X_2 must satisfy

$$X_1 \le 100$$
$$X_2 \le 50$$

The capacity constraints are a little more work. Since there are four machines, which run at most 2,400 minutes per week, we must ensure that our production quantities do not violate this constraint on each machine. Consider workstation A. Each unit of product 1 we produce requires 15 minutes on this workstation, while each unit of product 2 we produce requires 10 minutes. Hence, the total number of minutes of time required on workstation A to produce X_1 units of product 1 and X_2 units of product 2 is[10]

$$15X_1 + 10X_2$$

so the capacity constraint for workstation A is

$$15X_1 + 10X_2 \le 2{,}400$$

Proceeding analogously for workstations B, C, and D, we can write the other capacity constraints as follows:

$$\begin{aligned} 15X_1 + 35X_2 &\le 2{,}400 \quad && \text{workstation B} \\ 15X_1 + 5X_2 &\le 2{,}400 && \text{workstation C} \\ 25X_1 + 14X_2 &\le 2{,}400 && \text{workstation D} \end{aligned}$$

[9] Note that we are neglecting labor and overhead costs in our estimates of unit profit. This is reasonable if these costs are not affected by the choice of production quantities, that is, if we won't change the size of the workforce or the number of machines in the shop.

[10] Note that this constraint does not address such detailed considerations as setup times that depend on the sequence of products run on workstation A or whether full utilization of workstation A is possible given the WIP in the system. But as we discussed in Chapter 13, these issues are addressed at a lower level in the production planning and control hierarchy (e.g., in the sequencing and scheduling module).

We have now completely defined the following LP model of our optimization problem:

$$\text{Maximize} \quad 45X_1 + 60X_2 \tag{16.107}$$

Subject to:

$$X_1 \le 100 \tag{16.108}$$

$$X_2 \le 50 \tag{16.109}$$

$$15X_1 + 10X_2 \le 2{,}400 \tag{16.110}$$

$$15X_1 + 35X_2 \le 2{,}400 \tag{16.111}$$

$$15X_1 + 5X_2 \le 2{,}400 \tag{16.112}$$

$$25X_1 + 14X_2 \le 2{,}400 \tag{16.113}$$

Some LP packages allow the user to enter the problem in a form almost identical to that shown in formulation (16.107)–(16.113). Spreadsheet programs generally require the decision variables to be entered into cells and the constraints specified in terms of these cells. More sophisticated LP solvers allow the user to specify blocks of similar constraints in a concise form, which can substantially reduce modeling time for large problems.

Finally, with regard to formulation, we point out that we have not stated explicitly the constraints that X_1 and X_2 be nonnegative. Of course, they must be, since negative production quantities make no sense. In many LP packages, decision variables are assumed to be nonnegative unless the user specifies otherwise. In other packages, the user must include the nonnegativity constraints explicitly. This is something to beware of when using LP software.

Solution

To get a general idea of how an LP package works, let us consider the above formulation from a mathematical perspective. First, note that any pair of X_1 and X_2 that satisfies

$$15X_1 + 35X_2 \le 2{,}400 \quad \text{workstation B}$$

will also satisfy

$$15X_1 + 10X_2 \le 2{,}400 \quad \text{workstation A}$$

$$15X_1 + 5X_2 \le 2{,}400 \quad \text{workstation C}$$

because these differ only by having smaller coefficients for X_2. This means that the constraints for workstations A and C are redundant. Leaving them out will not affect the solution. In general, it does not hurt anything to have redundant constraints in an LP formulation. But to make our graphical illustration of how LP works as clear as possible, we will omit constraints (16.110) and (16.112) from here on.

Figure 16.15 illustrates problem (16.107)–(16.113) in graphical form, where X_1 is plotted on the horizontal axis and X_2 is plotted on the vertical axis. The shaded area is the **feasible region,** consisting of all the pairs of X_1 and X_2 that satisfy the constraints. For instance, the demand constraints (16.108) and (16.109) simply state that X_1 cannot be larger than 100, and X_2 cannot be larger than 50. The capacity constraints are graphed by noting that, with a bit of algebra, we can write constraints (16.111) and (16.113) as

$$X_2 \le -\left(\frac{15}{35}\right)X_1 + \frac{2{,}400}{35} = -0.429X_1 + 68.57 \tag{16.114}$$

$$X_2 \le -\left(\frac{25}{14}\right)X_1 + \frac{2{,}400}{14} = -1.786X_1 + 171.43 \tag{16.115}$$

If we replace the inequalities with equality signs in Equations (16.114) and (16.115), then these are simply equations of straight lines. Figure 16.15 plots these lines. The set of X_1 and X_2 points that satisfy these constraints is all the points lying below both of these lines. The points marked by the shaded area are those satisfying all the demand, capacity, and nonnegativity constraints. This type of feasible region defined by linear constraints is known as a **polyhedron.**

Figure 16.15

Feasible region for LP example

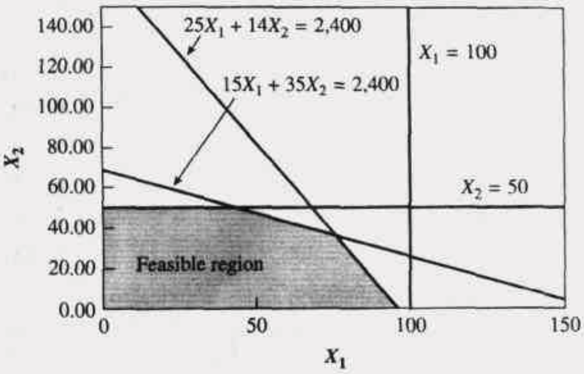
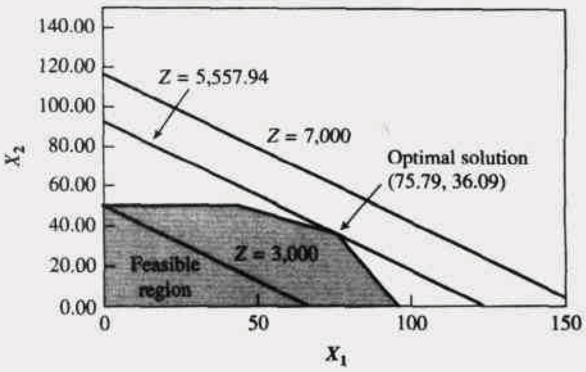


Figure 16.16

Solution to LP example



Now that we have characterized the feasible region, we turn to the objective. Let Z represent the value of the objective (i.e., net profit achieved by producing quantities X_1 and X_2). From objective (16.107), X_1 and X_2 are related to Z by

$$45X_1 + 60X_2 = Z \tag{16.116}$$

We can write this in the usual form for a straight line as

$$X_2 = \left(\frac{-45}{60}\right) X_1 + \frac{Z}{60} = -0.75X_1 + \frac{Z}{60} \tag{16.117}$$

Figure 16.16 illustrates Equation (16.117) for $Z = 3{,}000$, 5,557.94, and 7,000. Notice that for $Z = 3{,}000$, the line passes through the feasible region, leaving some points above it. Hence, we can feasibly increase profit (that is, Z). For $Z = 7{,}000$ the line lies entirely above the feasible region. Hence, $Z = 7{,}000$ is not feasible. For $Z = 5{,}557.94$, the objective function just touches the feasible region at a single point, the point ($X_1 = 75.79$, $X_2 = 36.09$). This is the **optimal solution.** Values of Z above 5,557.74 are infeasible, values below it are suboptimal. The optimal product mix, therefore, is to produce 75.79 (or 75, rounded to an integer value) units of product 1 and 36.09 (rounded to 36) units of product 2.

We can think of finding the solution to an LP by steadily increasing the objective value (Z), moving the objective function up and to the right, until it is just about to leave the feasible region. Because the feasible region is a polyhedron whose sides are made up of linear constraints, the last point of contact between the objective function and the feasible region will be a corner, or **extreme point,** of the feasible region.[11] This observation allows the optimization algorithm to ignore the infinitely many points inside the feasible region and search for a solution among the finite set of extreme points. The **simplex algorithm,** developed in the 1940s and still widely used, works in just this way, proceeding around the outside of the polyhedron, trying extreme points until an optimal one is found. Other, more modern algorithms use different schemes to find the optimal point, but will still converge to an extreme-point solution.

Sensitivity Analysis

The fact that the optimal solution to an LP lies at an extreme point enables us to perform useful sensitivity analysis on the optimal solution. The principal sensitivity information available to us falls into the following three categories.

[11] Actually, it is possible that the optimal objective function lies right along a flat spot connecting two extreme points of the polyhedron. When this occurs, there are many pairs of X_1 and X_2 that attain the optimal value of Z, and the solution is called **degenerate.** Even in this case, however, an extreme point (actually, at least two extreme points) will be among the optimal solutions.

1. **Coefficients in the objective function.** For instance, if we were to change the unit profit for product 1 from \$45 to \$60, then the equation for the objective function would change from Equation (16.117) to

$$X_2 = \left(-\frac{60}{60}\right) X_1 + \frac{Z}{60} = -X_1 + \frac{Z}{60} \qquad (16.118)$$

so the slope changes form -0.75 to -1; that is, it gets steeper. Figure 16.17 illustrates the effect. Under this change, the optimal solution remains ($X_1 = 75.79$, $X_2 = 36.09$). Note, however that while the decision variables remain the same, the objective function does not. When the unit profit for product 1 increases to \$60, the profit becomes

$$60(75.79) + 60(36.09) = \$6{,}712.80$$

The optimal decision variables remain unchanged until the coefficient of X_1 in the objective function reaches 107.14. When this happens, the slope becomes so steep that the point where the objective function just touches the feasible region moves to the extreme point ($X_1 = 96$, $X_2 = 0$). Geometrically, the objective function "rocked around" to a new extreme point. Economically, the profit from product 1 reached a point where it became optimal to produce all product 1 and no product 2.

In general, LP packages will report a range for each coefficient in the objective function for which the optimal solution (in terms of the decision variables) remains unchanged. Note that these ranges are valid only for one-at-a-time changes. If two or more coefficients are changed, the effect is more difficult to characterize. One has to rerun the model with multiple coefficient changes to get a feel for their effect.

2. **Coefficients in the constraints.** If the number of minutes required on workstation B by product 1 is changed from 15 to 20, then the equation defined by the capacity constraint for workstation B changes from Equation (16.114) to

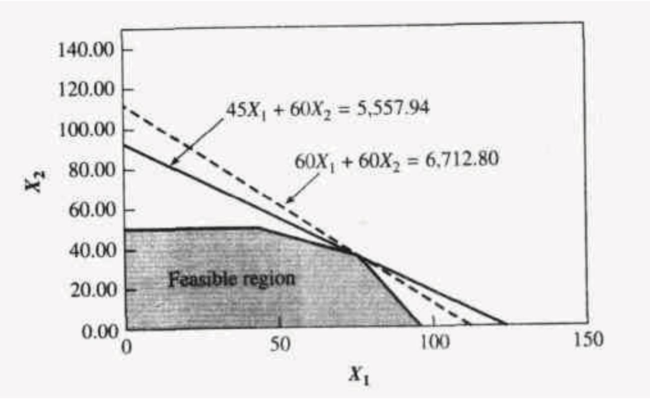
$$X_2 \le -\left(\frac{20}{35}\right) X_1 + \frac{2{,}400}{35} = -0.571X_1 + 68.57 \qquad (16.119)$$

so the slope changes from -0.429 to -0.571; again, it becomes steeper. In a manner analogous to that described above for coefficients in the objective function, LP packages can determine how much a given coefficient can change before it ceases to define the optimal extreme point. However, because changing the coefficients in the constraints moves the extreme points themselves, the optimal decision variables will also change. For this reason, most LP packages do not report this sensitivity data, but rather make use of this product as part of a **parametric programming** option to quickly generate new solutions for specified changes in the constraint coefficients.

3. **Right-hand side coefficients.** Probably the most useful sensitivity information provided by LP models is for the right-hand side variables in the constraints. For instance, in formulation (16.107)–(16.113), if we run 100 minutes of overtime per week on machine B, then its right-hand

FIGURE 16.17

Effect of changing objective coefficients LP example



side will increase from 2,400 to 2,500. Since this is something we might want to consider, we would like to be able to determine its effect. We do this differently for two types of constraints:
a. **Slack constraints** are constraints that do not define the optimal extreme point. The capacity constraints for workstations A and C are slack, since we determined right at the outset that they could not affect the solution. The constraint $X_2 \leq 50$ is also slack, as can be seen in Figures 16.15 and 16.16, although we did not know this until we solved the problem.

Small changes in slack constraints do not change the optimal decision variables or objective value at all. If we change the demand constraint on product 2 to $X_2 \leq 49$, it still won't affect the optimal solution. Indeed, not until we reduce the constraint to $X_2 \leq 36.09$ will it have any effect. Likewise, increasing the right-hand side of this constraint (above 50) will not affect the solution. Thus, for a slack constraint, the LP package tells us how far we can vary the right-hand side without changing the solution. These are referred to as the **allowable increase** and **allowable decrease** of the right-hand side coefficients.
b. **Tight constraints** are constraints that define the optimal extreme point. Changing them changes the extreme point, and hence the optimal solution. For instance, the constraint that the number of hours per week on workstation B not exceed 2,400, that is,

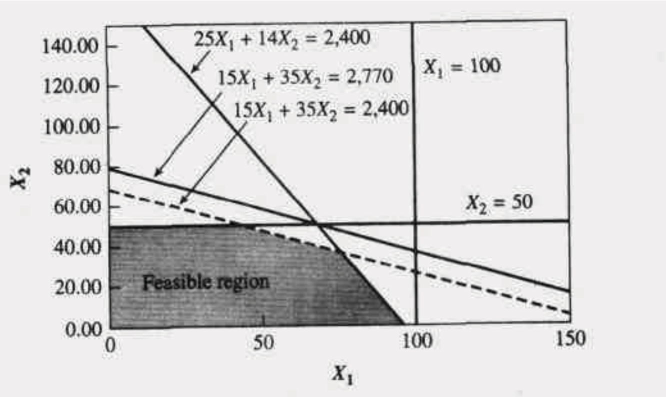
$$15X_1 + 35X_2 \leq 2{,}400$$

is a tight constraint in Figures 16.15 and 16.16. If we increase or decrease the right-hand side, the optimal solution will change. However, if the changes are small enough, then the optimal extreme point will still be defined by the same constraints (i.e., the time on workstations B and D). Because of this, we are able to compute the following:

> **Shadow prices** are the amount by which the objective increases per unit increase in the right-hand side of a constraint. Since slack constraints do not affect the optimal solution, changing their right-hand sides has no effect, and hence their shadow prices are always zero. Tight constraints, however, generally have nonzero shadow prices. For instance, the shadow price for the constraint on workstation B is 1.31. (Any LP solver will automatically compute this value.) This means that the objective will increase by \$1.31 for every extra minute per week on the workstation. So if we can work 2,500 minutes per week on workstation B, instead of 2,400, the objective will increase by $100 \times 1.31 = \$131$.
>
> **Maximum allowable increase/decrease** gives the range over which the shadow prices are valid. If we change a right-hand side by more than the maximum allowable increase or decrease, then the set of constraints that define the optimal extreme point may change, and hence the shadow price may also change. For example, as Figure 16.18 shows, if we increase the right-hand side of the constraint on workstation B from 2,400 to 2,770, the constraint moves to the very edge of the feasible region defined by $25X_1 + 14X_2 \leq 2{,}400$ (machine D) and $X_2 \leq 50$. Any further increases in the right-hand side will cause this constraint to become slack. Hence, the shadow price is \$1.31 up to a maximum allowable

FIGURE 16.18

Feasible region when I of constraint of workstation B is increa to 2,770



increase of 370 (that is, 2,770 − 2,400). In this example, the shadow price is zero for changes above the maximum allowable increase. This is not always the case, however, so in general we must resolve the LP to determine the shadow prices beyond the maximum allowable increase or decrease.

Study Questions

1. Although the technology for solving aggregate planning models (linear programming) is well established and AP modules are widely available in commercial systems (e.g., MRP II systems), aggregate planning does not occupy a central place in the planning function of many firms. Why do you think this is true? What difficulties in modeling, interpreting, and implementing AP models might be contributing to this?
2. Why does it make sense to consider workforce planning and aggregate planning simultaneously in many situations?
3. What is the difference between a **chase** production plan and a **level** production plan, with respect to the amount of inventory carried and the fluctuation in output quantity over time? How do the production plans generated by an LP model relate to these two types of plan?
4. In a basic LP formulation of the product mix aggregate planning problem, what information is provided by the following?
   a. The optimal decision variables.
   b. The optimal objective function.
   c. Identification of which constraints are tight and which are slack.
   d. Shadow prices for the right-hand sides of the constraints.

Problems

1. Suppose a plant can supplement its capacity by subcontracting part of or all the production of certain parts.
   a. Show how to modify LP (16.28)–(16.32) to include this option, where we define

   V_{it} = units of product i received from a subcontractor in period t
   k_{it} = premium paid for subcontracting product i in period t (i.e., cost above variable cost of making it in-house)
   $\underline{v}_{it}$ = minimum amount of product i that must be purchased in period t (e.g., specified as part of long-term contract with supplier)
   $\bar{v}_{it}$ = maximum amount of product i that can be purchased in period t (e.g., due to capacity constraints on supplier, as specified in long-term contract)

   b. How would you modify the formulation in part a if the contract with a supplier stipulated only that total purchases of product i over the time horizon must be at least $\underline{v}_i$?
   c. How would you modify the formulation in part a if the supplier contract, instead of specifying $\underline{v}$ and $\bar{v}$, stipulated that the firm specify a base amount of product i, to be purchased every month, and that the maximum purchase in a given month can exceed the base amount by no more than 20 percent?
   d. What role might models like those in parts a to c play in the process of negotiating contracts with suppliers?
2. Show how to modify LP (16.49)–(16.54) to represent the case where overtime on all the workstations must be scheduled simultaneously (i.e., if one resource runs overtime, all resources run overtime). Describe how you would handle the case where, in general, different workstations can have different amounts of overtime, but two workstations, say A and B, must always be scheduled for overtime together.

3. Show how to modify LP (16.61)–(16.67) of the workforce planning problem to accommodate multiple products.

4. You have just been made corporate vice president in charge of manufacturing for an automotive components company and are directly in charge of assigning products to plants. Among many other products, the firm makes automotive batteries in three grades: heavy-duty, standard, and economy. The unit net profits and maximum daily demand for these products are given in the first table below. The firm has three locations where the batteries can be produced. The maximum assembly capacities, for any mix of battery grades, are given in the second table below. The number of batteries that can be produced at a location is limited by the amount of suitably formulated lead the location can produce. The lead requirements for each grade of battery and the maximum lead production for each location are also given in the following tables.

Product	Unit Profit ($/battery)	Maximum Demand (batteries/day)	Lead Requirements (lbs/battery)
Heavy-duty	12	700	21
Standard	10	900	17
Economy	7	450	14

Plant Location	Assembly Capacity (batteries/day)	Maximum Lead Production (lbs/day)
1	550	10,000
2	750	7,000
3	225	4,200

a. Formulate a linear program that allocates production of the three grades among the three locations in a manner that maximizes profit.

b. Suppose company policy requires that the fraction of capacity (units scheduled/assembly capacity) be the same at all locations. Show how to modify your LP to incorporate this constraint.

c. Suppose company policy dictates that at least 50 percent of the batteries produced must be heavy-duty. Show how to modify your LP to incorporate this constraint.

5. Youohimga, Inc., makes a variety of computer storage devices, which can be divided into two main families that we call A and B. All devices in family A have the same routing and similar processing requirements at each workstation; similarly for family B. There are a total of 10 machines used to produce the two families, where the routings for A and B have some workstations in common (i.e., shared) but also contain unique (unshared) workstations.

Because Youohimga does not always have sufficient capacity to meet demand, especially during the peak demand period (i.e., the months near the start of the school year in September), in the past it has contracted out production of some of its products to vendors (i.e., the vendors manufacture devices that are shipped out under Youohimga's label). This year, Youohimga has decided to use a systematic aggregate planning process to determine vendoring needs and a long-term production plan.

a. Using the following notation

X_{it} = units of family i (i = A, B) produced in month t ($t = 1, \ldots, 24$) and available to meet demand in month t

V_{it} = units of family i purchased from vendor in month t and available to meet demand in month t

I_{it} = finished goods inventory of family i at end of month t

d_{it} = units of family i demanded (and shipped) during month t

c_{jt} = hours available on work center $j\,(j = 1, \ldots, 10)$ in month t

a_{ij} = hours required at work center j per unit of family i

v_i = premium (i.e., extra cost) per unit of family i that is vendored instead of being produced in-house

h_i = holding cost to carry one unit of family i in inventory from one month to the next

formulate a linear program that minimizes the cost (holding plus vendoring premium) over a two-year (24-month) planning horizon of meeting monthly demand (i.e., no backorders are permitted). You may assume that vendor capacity for both families is unlimited and that there is no inventory of either family on hand at the beginning of the planning horizon.

b. Which of the following factors might make sense to examine in the aggregate planning model to help formulate a sensible vendoring strategy?

- Altering machine capacities
- Sequencing and scheduling
- Varying size of workforce
- Alternate shop floor control mechanisms
- Vendoring individual operations rather than complete products
- All the above

c. Suppose you run the model in part *a* and it suggests vendoring 50 percent of the total demand for family A and 50 percent of the demand for B. Vendoring 100 percent of A and 0 percent of B is capacity-feasible, but results in a higher cost in the model. Could the 100–0 plan be preferable to the 50–50 plan in practice? If so, explain why.

6. Mr. B. O'Problem of Rancid Industries must decide on a production strategy for two top-secret products, which for security reasons we will call X and Y. The questions concern (1) whether to produce these products at all and (2) how much of each to produce. Both products can be produced on a single machine, and there are three brands of machine that can be leased for this purpose. However, because of availability problems, Rancid can lease at most one of each brand of machine. Thus, O'Problem must also decide which, if any, of the machines to lease. The relevant machine and product data are given below:

Machine	**Hours to Produce One Unit of X**	**Hours to Produce One Unit of Y**	**Weekly Capacity (hours)**	**Weekly Lease + Operating Cost ($)**
Brand 1	0.5	1.2	80	20,000
Brand 2	0.4	1.2	80	22,000
Brand 3	0.6	0.8	80	18,000

Product	**Maximum Demand (units/week)**	**Net Unit Profit ($/unit)**
X	200	150
Y	100	225

a. Letting X_{ij} represent the number of units of product i produced per week on machine j (for example, X_{A1} is the number of units of A produced on the brand 1 machine), formulate an LP to maximize weekly profit (including leasing cost) subject to the capacity and demand constraints. (*Hint:* Observe that the leasing/operating cost for a particular machine is only incurred if that machine is used and that this cost is fixed for any nonzero production level. Carefully define 0–1 integer variables to represent the all-or-nothing aspects of this decision.)

b. Suppose that the suppliers of brand 1 machines and brand 2 machines are feuding and will not service the same company. Show how to modify your formulation to ensure that Rancid leases either brand 1 or brand 2 or neither, but not both.

7. All-Balsa, Inc., produces two models of bookcases, for which the relevant data are summarized as follows:

	Bookcase 1	**Bookcase 2**
Selling price	\$15	\$8
Labor required	0.75 hr/unit	0.5 hr/unit
Bottleneck machine time required	1.5 hr/unit	0.8 hr/unit
Raw material required	2 bf/unit	1 bf/unit

$P1$ = units of bookcase 1 produced per week

$P2$ = units of bookcase 2 produced per week

OT = hours of overtime used per week

RM = board-feet of raw material purchased per week

$A1$ = dollars per week spent on advertising bookcase 1

$A2$ = dollars per week spent on advertising bookcase 2

Each week, up to 400 board feet (bf) of raw material is available at a cost of \$1.50/bf. The company employs four workers, who work 40 hours per week for a total regular time labor supply of 160 hours per week. They work regardless of production volumes, so their salaries are treated as a fixed cost. Workers can be asked to work overtime and are paid \$6 per hour for overtime work. There are 320 hours per week available on the bottleneck machine.

In the absence of advertising, 50 units per week of bookcase 1 and 60 units per week of bookcase 2 will be demanded. Advertising can be used to stimulate demand for each product. Experience shows that each dollar spent on advertising bookcase 1 increases demand for bookcase 1 by 10 units, while each dollar spent on advertising bookcase 2 increases demand for bookcase 2 by 15 units. At most, \$100 per week can be spent on advertising.

An LP formulation and solution of the problem to determine how much of each product to produce each week, how much raw material to buy, how much overtime to use, and how much advertising to buy are given below. Answer the following on the basis of this output.

```
MAX      15 P1 + 8 P2 - 6 OT - 1.5 RM - A1 - A2
SUBJECT TO
        2)    P1 - 10 A1 <=   50
        3)    P2 - 15 A2 <=   60
        4)    0.75 P1 + 0.5 P2 - OT <=    160
        5)    2 P1 + P2 - RM <=   0
        6)    RM <=   400
        7)    A1 + A2 <=    100
        8)    1.5 P1 + 0.8 P2 <=   320
END
```

```
          OBJECTIVE FUNCTION VALUE

          1)    2427.66700

    VARIABLE        VALUE          REDUCED COST
          P1       160.000000          .000000
          P2        80.000000          .000000
          OT          .000000         2.133334
          RM       400.000000          .000000
          A1        11.000000          .000000
          A2         1.333333          .000000

         ROW   SLACK OR SURPLUS    DUAL PRICES
          2)          .000000          .100000
          3)          .000000          .066667
          4)          .000000         3.866666
          5)          .000000         6.000000
          6)          .000000         4.500000
          7)        87.666660          .000000
          8)        16.000000          .000000

   NO. ITERATIONS=        5

   RANGES IN WHICH THE BASIS IS UNCHANGED:

                            OBJ COEFFICIENT RANGES
   VARIABLE          CURRENT        ALLOWABLE        ALLOWABLE
                      COEF          INCREASE         DECREASE
         P1        15.000000         .966667          .533333
         P2         8.000000         .266667          .483333
         OT        -6.000000        2.133334         INFINITY
         RM        -1.500000        INFINITY         4.500000
         A1        -1.000000        1.000000         5.333335
         A2        -1.000000        1.000000         7.249999

                            RIGHT-HAND SIDE RANGES
        ROW          CURRENT       ALLOWABLE        ALLOWABLE
                       RHS          INCREASE         DECREASE
          2        50.000000      110.000000       876.666600
          3        60.000000       20.000000      1315.000000
          4       160.000000       27.500000         2.500000
          5          .000000        6.666667        55.000000
          6       400.000000        6.666667        55.000000
          7       100.000000        INFINITY        87.666660
          8       320.000000        INFINITY        16.000000
```

a. If overtime costs only $4 per hour (and all other parameters remained unchanged), how much overtime should All-Balsa use?

b. If each unit of bookcase 1 sold for $15.50 (and all other parameters are unchanged), what will the optimal profit per week be—or can you not tell without resolving the LP?

c. What is the most All-Balsa should be willing to pay for another unit of raw material?

d. If each worker were required (as part of the regular workweek) to work 45 hours per week (and all other parameters remained unchanged), what would the company's profit be?

e. If each unit of bookcase 2 sold for $10 (and all other parameters remained unchanged), what would be the optimal quantity of bookcase 2 to produce—or can you not tell without resolving the LP?

f. Reconsider the All-Balsa problem formulation and suppose that instead of having 400 bf of raw material available at \$1.50/bf, All-Balsa faces a two-tier pricing scheme such that the first 200 bf/week costs \$2.00/bf, but any amount above 200 bf/week up to a limit of an additional 300 bf/week costs \$$p$/bf. (*Note: p is a constant,* not a variable, and we cannot purchase the \$$p$/bf raw material unless we first purchase 200 bf of the \$2.00 raw material.) To modify the LP to compute an "optimal" production/advertising policy, we define

RM1 = bf of raw material purchased at \$2.00/bf
RM2 = bf of raw material purchased at \$$p$/bf

To formulate an appropriate LP to represent this new pricing scheme, we first replace 1.5RM in the objective function by 2RM1 + pRM2.

i. If $p > 2$, what other changes in the previous LP make it properly reflect the new pricing scheme?
ii. If $p < 2$, what other changes in the previous LP make it properly reflect the new pricing scheme?

8. Consider a production line with four workstations, labeled $j = 1, 2, 3$, and 4, in tandem (all products flow through all four machines in order). Three different products, labeled i = A, B, and C, are produced on the line. The hours required on each workstation for each product and the net profit per unit sold (r_i) are given as follows:

	j				
i	1	2	3	4	r_i
A	2.4	1.1	0.8	3.0	\$50
B	2.0	2.2	1.2	2.1	\$65
C	0.9	0.9	1.0	2.5	\$70

The number of hours available (c_{jt}) and the upper and lower limits on demand ($\bar{d}_{it}$ and $\underline{d}_{it}$) for each product over the next four quarters are as follows:

t	1	2	3	4
c_{1t}	640	640	1,280	1,280
c_{2t}	640	640	640	640
c_{3t}	1,920	1,920	1,920	1,920
c_{4t}	1,280	1,280	1,280	2,560
$\bar{d}_{At}$	100	50	50	75
$\underline{d}_{At}$	0	0	0	0
$\bar{d}_{Bt}$	100	100	100	100
$\underline{d}_{Bt}$	20	20	20	25
$\bar{d}_{Ct}$	300	250	250	400
$\underline{d}_{Ct}$	0	0	0	50

a. Suppose we use a quarterly holding cost of \$5 and a quarterly backorder cost of \$10 per item on all products and allow backordering. Formulate an LP to maximize profit minus holding and backorder costs subject to the constraints on workstation capacity and minimum/maximum sales.

b. Using the LP solver of your choice, solve your formulation in part *a*. Which constraints are binding in your solution?

c. Suppose that there is an inspect operation immediately after station 2 (which has plenty of capacity and therefore does not need to be modeled as an extra resource) and 20 percent of the parts (regardless of product type) are recycled back through stations 1 and 2. Show how to modify your formulation in part *a* to model this.

9. A manufacturer of high-voltage switches projects demand (in units) for the upcoming year to be as follows.

Jan	1,000	Jul	3,200
Feb	1,000	Aug	2,000
Mar	1,000	Sep	1,000
Apr	2,000	Oct	900
May	2,400	Nov	800
Jun	2,500	Dec	800

The plant runs 160 hours per month and produces at an average rate of 10 switches per hour. Unit profit per switch sold is $50, and the estimated cost to hold a switch in inventory for one month is $5. There is no inventory at the start of the year. Overtime can be used at a cost of $300 per hour.

a. Compute the inventory-holding and overtime cost of a chase production strategy (i.e., producing the amount demanded in each month).

b. Compute the inventory holding and overtime cost of a level production strategy (i.e., producing the same amount each month). If the monthly production quantity is set equal to average monthly demand, how much inventory will be left at the end of the year?

c. Compute a production strategy by solving a linear program to maximize profit (i.e., net sales revenue minus inventory carrying cost minus overtime cost). Is the amount of overtime in the plan reasonable? If not, what changes to the LP model could be made to generate a more reasonable solution?

d. How does the solution change if the inventory carrying cost is reduced to $3 per unit per month? If overtime costs are reduced to $200 per hour? Given that these costs are approximate, what do these results imply about the production plan?

10. Reconsider Problem 2 of Chapter 6 in which a manufacturer produced three models of vacuum cleaner on a three-station production line.

a. Use linear programming to compute a monthly production plan that maximizes monthly profit, and compare it to the profit resulting from the current plan given in Chapter 6 and those suggested by the labor hours and ABA cost accounting calculations.

b. Could this LP solution have been arrived at by rank-ordering the products according to profitability by a cost accounting scheme? What does this say about the effectiveness of using accounting methods to plan production schedules?

CHAPTER

17 Supply Chain Management

One's work may be finished some day,
but one's education never.
Alexandre Dumas

17.1 Introduction

A major theme of this book is the central role of inventory in the operational behavior of a production system. We began with a historical review of inventory control and its relationship to production control in Part I. In Part II, we deepened our understanding of the interaction between inventory (WIP, in particular) and other performance measures, such as throughput and cycle time. Now in Part III we are ready to combine our historical and factory physics insights to address the practical problem of managing inventories in a manufacturing system. Our objective is to improve inventory *efficiency* throughout the system. That is, we do not simply seek to reduce inventories; we seek to ensure that the purpose of inventories is met with minimal dollar investment. In modern parlance, this overall systemwide coordination of inventory stocks and flows is known as **supply chain management.**

For purposes of our discussions here, we divide inventories in a supply chain into four categories:

1. **Raw materials** are components, subassemblies, or materials that are purchased from outside the plant and used in the fabrication/assembly processes inside the plant.
2. **Work in process (WIP)** includes all unfinished parts or products that have been released to a production line.
3. **Finished goods inventory (FGI)** is finished product that has not been sold.
4. **Spare parts** are components that are used to maintain or repair production equipment.

The reasons for holding each of these types of inventory, and therefore the options for improving efficiency, are different. Hence, we treat each category separately in the following discussions.

17.2 Reasons for Holding Inventory

17.2.1 Raw Materials

If we could receive raw materials from suppliers in literal just-in-time fashion (i.e., exactly when needed by the production system), we would not need to carry any raw materials inventories. Since this is never possible in practice, all manufacturing systems carry stocks of raw materials. There are three main factors that influence the size of these stocks.

1. **Batching.** Quantity discounts from suppliers, limited capacity of the plant's purchasing function (e.g., a limit on the number of purchase orders that can be placed and tracked), and economies of scale in deliveries provide incentive to order raw materials in bulk.[1] We refer to inventory that addresses batching considerations as **cycle stock,** since it represents stock held between ordering cycles.
2. **Variability.** When production gets ahead of schedule, supplier deliveries get behind schedule, or quality problems cause excessive scrap loss, the line will shut down for lack of materials if extra stock is not available. This extra stock can be planned for directly as a **safety stock** (i.e., by ordering so that expected stock levels remain above the safety level) or be the consequence of a **safety lead time** (i.e., order materials so that they arrive before needed and therefore wait in raw materials inventory). In either case, we refer to inventory carried as protection against variability as safety stock.
3. **Obsolescence.** Changes in demand or design can render some materials no longer needed, so some inventory in manufacturing systems does not address either of the above purposes. This inventory, which we term **obsolete inventory,** may have been ordered as cycle or safety stock, but is now essentially useless and must be disposed of and written off as quickly as financial reporting considerations will permit.

To recognize these reasons for carrying raw materials inventories is useful in identifying improved management policies. However, one should remember that they are not strictly separate. For instance, as we pointed out in Chapter 2, safety stock *and* cycle stock provide protection against variability (i.e., because if we order in very large batches, then we reduce the frequency with which inventory levels fall to the point where a stockout is possible). Also, the level of obsolete inventory is clearly affected by the levels of cycle and safety stock (i.e., if we order in large batches or carry large safety stocks, then we risk having large amounts of inventory become obsolete due to system changes). Appreciating these interactions can also help us devise raw materials management policies.

17.2.2 Work in Process

Despite the JIT goal of zero inventories, we can never operate a manufacturing system with zero WIP since, as we saw in Part II, zero WIP implies zero throughput. In Chapter 7, we derived a **critical WIP** level that represents the smallest WIP level required by

[1] These factors are precisely those that motivated the fixed order cost in the EOQ model presented in Chapter 2. The EOQ model balances this fixed cost against inventory carrying costs to determine an economic order quantity.

a line to achieve full throughput under the best conditions. Under realistic conditions, actual WIP levels frequently exceed the critical WIP level by a large amount (e.g., often 20 to 30 times). This WIP will be in one of five states:

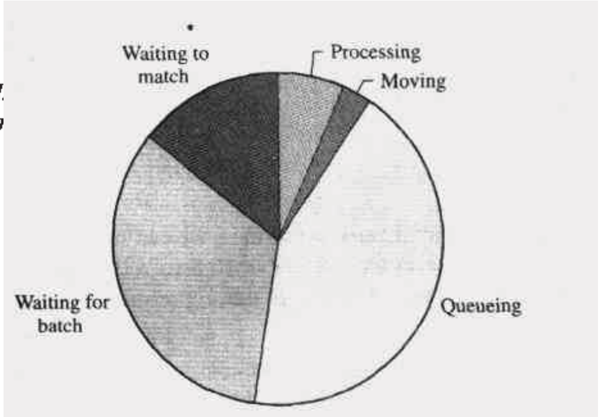
1. **Queueing** if it is waiting for a resource (person, machine, or transport device).
2. **Processing** if it is being worked on by a resource.
3. **Waiting for batch** if it has to wait for other jobs to arrive in order to form a batch. This batch may serve to fill a bulk manufacturing operation (e.g., heat treat, in which a roomful of jobs is subjected to a burn-in operation simultaneously) or a move operation (e.g., when jobs are moved only in full pallets). Note that once the process or move batch has been formed, any additional waiting time for the resource (e.g., for the heat treater or the forklift to become available) is classed as queueing time.
4. **Moving** if it is actually being transported between resources.
5. **Waiting to match** if it consists of components waiting at an assembly operation for their counterparts to arrive so that an assembly can occur. Once the entire "kit" of parts has arrived, any additional waiting time for the assembly resource is defined as queueing time.

To use the above classification in a WIP management/reduction program, two observations are needed. First, as illustrated in Figure 17.1, in most manufacturing systems the fraction of WIP that is actually processing or moving is small (e.g., less than 10 percent; see Bradt 1983 for empirical documentation). The majority of WIP is in queue, waiting for batch, or waiting to match. Clearly, a WIP reduction program must address these latter categories to be successful.

Second, queueing WIP, wait-for-batch WIP, and wait-to-match WIP are the result of different causes. As we saw in Part II, the principal causes of queueing are high utilization and variability (both flow variability and process variability). Wait-for-batch WIP is clearly caused by batching for process or transport; the larger the batch size, the more WIP required. Wait-to-match WIP is caused by lack of synchronization in the arrival of parts to the assembly process, some of which is due to simple flow variability and some of which can be caused by the production control process. These differences imply that the different types of WIP are amenable to different management policies, as we will discuss later.

FIGURE 17.1

Typical breakdown of WI in a manufacturing syste



17.2.3 Finished Goods Inventory

If we could ship everything we produced directly to customers as soon as processing was complete, there would be no need for FGI. Although some manufacturing systems (e.g., heavily loaded job shops that make custom products) can almost achieve this, many cannot. There are five basic reasons for carrying FGI.

1. **Customer responsiveness.** To provide delivery lead times that are shorter than manufacturing cycle times, many firms make use of a **make-to-stock** (instead of a **make-to-order**) policy. For example, many products, such as building materials (e.g., roofing shingles, lumber), standard electrical components (e.g., resistors, capacitors), and basic food products (e.g., baking soda, corn oil) are **commodity** products. As such, their price and specifications (e.g., quality) are set by the market. The only competitive issue, then, is delivery. For this reason, such products are frequently produced to stock. The amount of FGI needed to support a given make-to-stock system depends on the variability of customer demand and the desired level of customer service.

An approach that combines the effectiveness of make-to-stock and make-to-order procedures is **assemble-to-order.** This procedure produces components to stock and then assembles these components to order. In the terminology of Chapter 10, make-to-order places the **push/pull interface** at raw materials, make-to-stock is places it at finished goods, while assemble-to-order places it somewhere in between. The result is faster response than the traditional make-to-order approach with less inventory than a make-to-stock policy.

2. **Batch production.** If, for whatever reason, production occurs in prespecified quantities (batches), then output will sometimes not match customer orders and any excess will go into finished goods inventory. For example, a steel mill that runs 250-ton batches (in order to efficiently utilize the casting furnace) but has customer orders averaging 50 tons will frequently have to place remnants of batches of various grades of steel into FGI.

3. **Forecast errors.** When jobs are released without firm customer orders, either to replenish stock in a make-to-stock system or to meet anticipated orders in a make-to-order system, product will inevitably be built that does not sell as anticipated. This excess will wind up in FGI.

4. **Production variability.** In a make-to-order system where orders cannot be shipped early (or have a limit on how early they can be shipped), variability in production *timing* will sometimes result in product that will have to reside in FGI while awaiting shipment. In either a make-to-order or a make-to-stock system, variability in production *quantity* (e.g., due to random yield loss) can result in overproduction relative to demand (e.g., if we "overinflate" to compensate for the yield loss). Again, the excess will go into FGI.

5. **Seasonality.** One approach to dealing with demand that varies with season (e.g., lawnmowers, snowblowers, room air conditioners) is to build inventory during the off season to meet peak demand. This **built-ahead inventory** will become part of FGI.

Notice that the factors motivating finished goods inventory interact. For instance, whenever we build FGI to provide short lead times or to cover seasonal demand we increase exposure of the system to forecasting errors. Because of this, it is important to view FGI holistically. Only by doing this can we consider basic structural changes that may offer significant potential. For instance, maybe the system should really be run in make-to-order instead of make-to-stock fashion; maybe excess capacity or seasonal labor should be used instead of built-ahead inventory to address seasonal demand, or maybe the push-pull interface should be relocated (e.g., to use an assemble-to-order strategy). We will return to these options in our discussion of improvement strategies.

17.2.4 Spare Parts

Spare parts are not used as direct inputs to finished products, but they do support the production process by keeping the machines running. In many systems the dollar value of inventory involved is not large, but the consequences of shortfalls can be severe (e.g., the entire line can be shut down for lack of a critical part). In some systems (e.g., a contract service operation that supports repairs in a nationwide network of machines), however, the dollar value of spare parts inventories can be substantial. In either case, the primary reasons for stocking spare parts are

1. **Service.** The main objective of any spare parts system is to support a maintenance and repair process. If repair personnel must wait for a part (e.g., from a central storage site or an outside supplier), then the time to complete a repair can be dramatically lengthened. All other things being equal, achieving higher service (i.e., avoidance of delay due to an out-of-stock part) requires a higher level of spare parts inventory.
2. **Purchasing/production lead times.** If spare parts could be purchased or produced instantly, there would be no need to stock them. Unfortunately, this is virtually never the case; so to provide the desired service, we must carry spare parts inventories. In general, the longer the lead time to obtain a part, the more stock we will have to carry.
3. **Batch replenishment.** If there are economies of scale in replenishing spare parts (e.g., quantity discounts on a purchased part or a large fixed cost to produce a part), then it may make sense to purchase them in bulk. Of course, a larger replenishment batch implies a higher average inventory level.

In theory, spare parts inventory systems are not much different from FGI systems. In both, we stock parts, possibly in batches, to satisfy an uncertain demand process with some level of service. Because of this similarity, it may well be possible to use similar tools for controlling spare parts and FGI. However, it is important to recognize the difference between the roles played by the two types of inventory. For instance, it may be reasonable to set a fill rate of 90 percent for FGI, based on industry benchmarking, say. But a 90 percent fill rate for spare parts may be far too low when one considers the logistical and financial consequences of causing a long machine outage by stocking out on a critical part. Thus, while we might use similar models to address the two types of inventory, we must carefully consider the costs and objectives involved in order to set appropriate parameters for the models.

Having reviewed the reasons for holding different types of inventory, we now review techniques for improving the efficiency (i.e., attaining the same benefits with a smaller overall investment) of each type of inventory.

17.3 Managing Raw Materials

As noted above, the objective in managing raw materials is to have them available when needed by the production process without carrying any more inventory than necessary. Some strategies can enhance our ability to do this for all parts. Others are economically viable for only certain classes of parts. Therefore, our basic strategy is one of "divide and conquer," in which we apply different approaches to different classes of raw material. In the following sections we present some overall improvement strategies, a classification scheme, and focused control policies geared to specific part classes.

17.3.1 Visibility Improvements

Obviously, we can do a better job of purchasing raw materials if we know what parts are needed than if we must guess. Unfortunately, manufacturing cycle times and purchasing lead times are frequently long enough to require us to purchase at least some of the materials before we have firm customer orders. In the short term, we may have no option other than to maintain safety stocks of raw materials to buffer against purchasing mistakes. In the long term, however, we can improve the situation via the following policies:

1. **Improve forecasting.** If forecasts of future demand are truly horrible, better projections may be possible through the use of systematic forecasting techniques (see Appendix 13A). However, such methods cannot get around the first law of forecasting—*forecasts are always wrong*. Thus, there are limits to the improvements possible through forecasting.
2. **Reduce cycle times.** Reduced manufacturing cycle times imply that jobs can be released closer to their due dates. Hence, purchased parts can be ordered later, when customer demands are firmer. In systems with long cycle times, cycle time reduction can improve forecasts much more than use of sophisticated forecasting techniques can. We discuss specific techniques for cycle time (and WIP) reduction in Section 17.4.
3. **Improve scheduling.** If scheduling is poor, then projected use of purchased parts may be very different from actual use. For instance, a schedule generated with an infinite-capacity MRP model may project much earlier completion of jobs than actually will occur. This will result in purchased parts arriving well before they are actually used and hence will cause raw materials inventories to be inflated. A good finite-capacity scheduler will generate more realistic schedules and thus will enable purchased parts to be brought in closer to when they are used.

17.3.2 ABC Classification

In most manufacturing systems, a small fraction of the purchased parts represent a large fraction of the purchasing expenditures.[2] To have maximum impact, therefore, management attention should be focused most closely on these parts. To accomplish this, many manufacturing firms use some sort of **ABC classification** for purchased parts and materials. In a typical definition of ABC categories, we rank-order the purchased parts according to the annual dollar value spent on each, and we define

A parts: the first 5 to 10 percent of the parts, accounting for 75 to 80 percent of total annual expenditures.

[2] This is an example of **Pareto's law,** commonly known as the "80-20 rule," named for Italian economist Vilfredo Pareto (1848–1923) who observed that a large fraction of wealth tends to be concentrated in a small fraction of the population.

B parts: the next 10 to 15 percent of the parts, accounting for 10 to 15 percent of total annual expenditures.

C parts: the bottom 80 percent or so of the parts, accounting for only 10 percent or so of total annual expenditures.

Because their number is relatively small and their cost is high, it makes sense to use sophisticated, time-consuming methods to tightly coordinate the arrival of A parts with their use by the production process. Such efforts are generally not warranted for C parts, since the cost of holding small excess quantities of inventory is not large. The B parts are in-between, so they deserve more attention than the C parts, but not as much as the A parts. Approaches may vary from system to system, but the main point of ABC classification remains the same: Inventories of different classes of parts should be treated differently.

We discuss some suitable techniques and where each is applicable in the following sections.

17.3.3 Just-in-Time

Very expensive A parts, for which holding inventory is costly, and extremely bulky parts (e.g., packaging materials), for which holding inventory is inconvenient, are good candidates for tight inventory control. The way to maintain the absolute minimum level of inventory of a part is to coordinate deliveries with use in the production process. This is precisely the idea behind just-in-time (JIT).

A typical JIT contract with a supplier calls for frequent deliveries (e.g., weekly, daily, or even more often, depending on the system) in small quantities closely matched to what is required by the production schedule. Since production schedules are prone to change, most JIT contracts allow adjustment of the order quantities almost up to the delivery time (although most contracts also specify limits on the amount of change allowed).

To give suppliers a reasonable chance of meeting delivery requirements, well-managed JIT procurement systems provide visibility of the production schedule to suppliers. The primary goal is to alert suppliers as quickly as possible to any changes in the schedule. But such visibility can have other benefits. It can eliminate the need for purchase orders. For instance, a contract with a supplier of automotive brakes might call for it to look at the final assembly schedule and deliver the proper brakes to support it. The system could go even further and eliminate invoices for the brakes by simply counting the number of automobiles produced and sending payment to the supplier for them. (The implicit, and reasonable, assumption is that every automobile has a set of brakes.)

In concept, JIT contracts with suppliers are very attractive. However, in order for them to work, suppliers must be reliable, with regard to both delivery timing and quality. If a shipment is late or defective, then the entire line may be stopped for lack of parts. Because of this, firms that rely extensively on JIT deliveries of raw materials generally institute some kind of **vendor certification** program. Good vendor certification programs involve both reviews of supplier procedures and efforts to help vendors improve their systems.

Because close supervision and cultivation of suppliers is a prerequisite for JIT deliveries of raw materials, this approach may not be a feasible option for smaller firms. A firm whose purchases compose a very small fraction of a supplier's business may simply lack the clout to persuade the supplier to deliver parts on a JIT basis. While the current trend toward responsiveness (e.g., as embodied in buzzwords such as *time-based competition, total cycle time, short-cycle manufacturing*) may be increasing the number of suppliers who are willing to offer JIT deliveries to firms other than their largest customers, true JIT contracts are still largely unavailable to the typical small firm. Thus, they must seek other approaches to managing expensive raw materials inventories.

17.3.4 Setting Safety Stock/Lead Times for Purchased Components

Even if a firm cannot or will not use JIT deliveries for expensive A parts, it still makes sense to link purchases of these parts closely to the production schedule (instead of, say, ordering infrequently in large batches and supplying the line from an amply stocked materials crib). In MRP language, this means that expensive parts should be ordered on a **lot-for-lot** basis. For example, if we plan to produce 1,000 high-resolution monitors n weeks from now, we should order 1,000 cathode-ray tubes to arrive some fixed safety lead time in advance of the schedule.[3]

Notice that this approach is different from JIT because we are ordering parts against a *planned* schedule, rather than having them delivered in synchronization with *actual* production. But if true JIT is not possible, this may be the best we can do. Of course, if (when) the schedule changes, production of the desired amounts may be impossible due to lack of appropriate raw materials. This implies that short delivery lead times are less difficult to work with than long ones, because purchases will be made closer to due dates, when the schedule consists more of firm orders and less of speculative forecasts. In the long run, a higher-priced supplier with short lead times may be more economical than a lower-priced one with long lead times.

As we noted in Chapter 12 in the context of supplier quality, management of purchased parts is extremely important in assembly systems with many parts. There we pointed out that if we purchase 10 parts with sufficient safety lead times such that each has a service level of 95 percent, then the probability of having all 10 parts arrive in time to meet the schedule is $0.95^{10} = 0.5987$, which represents very poor service. Assembly systems with many purchased parts require extremely high service for each part in order to meet schedules reliably. For instance, for all 10 parts to be available to meet the schedule 95 percent of the time requires that each part have a service level of $0.95^{1/10} = 0.9949$.

Finally, note that it is not necessary to set the same service level for every A part that is ordered on a lot-for-lot basis. If one part is particularly expensive, it might make sense to set its service relatively low (say, 96 percent) and the other service levels higher (say, 99.9 percent) to compensate. If we let S_j represent the service level chosen for the jth part and there are n parts in total, then we can ensure 95 percent compliance with the schedule provided we choose the S_j values such that

$$S_1 \cdot S_2 \cdots S_n = 0.95$$

A formal method for choosing service levels to meet an overall service level with minimal average investment in inventory is described in Hopp and Spearman (1993a).

17.3.5 Setting Order Frequencies for Purchased Components

The above JIT and lot-for-lot purchasing schemes are reasonable options for expensive A parts, and they might also work for intermediate B parts, but are generally not appropriate

[3] If yield loss is a problem, we may also need to maintain a planned level of safety stock.

for inexpensive C parts. It doesn't make sense to order screws, washers, two-cent resistors, etc., to be delivered in tight synchronization with the production schedule. The increased risk of an outage and the extra purchasing and material handling costs simply cannot be justified by reductions in inventory investment.

The problem of managing inexpensive purchased parts can be thought of in terms of **lot sizing.** The essential economic tradeoff is between inventory investment and purchasing cost. Recall that this is precisely the tradeoff addressed by the economic order quantity (EOQ) model. Indeed, we could directly apply the single-product model presented in Section 2.2, provided we are willing to ignore part interactions. That is, if we let

N = total number of distinct part numbers in system
D_j = demand rate (units per year) for part j
c_j = unit production cost of part j
A = fixed cost to place an order for any part
h_j = cost to hold one unit of part j for one year
Q_j = size of order or lot size for part j (decision variable)

we can compute the lot size for part j using the standard EOQ formula:

$$Q_j^* = \sqrt{\frac{2AD_j}{h_j}} \tag{17.1}$$

The most difficult input to estimate in this formula is the fixed order cost,[4] A. Ideally, this should reflect those costs that are incurred each time an order is placed. These could include actual shipping costs, purchasing agent time spent to process and follow up on the order, time required to receive the order, and so on. Overhead costs (e.g., maintenance of a purchasing department) should not be included in A_j.

A potential problem with the above approach is that it does not consider interactions between parts, which can occur when (1) parts share common delivery systems and (2) we consider the overall capacity of the purchasing department. For instance, if different parts can share common delivery trucks, then there is an incentive to order parts at the same time, when possible. In Chapter 2, we mentioned the powers-of-two replenishment policy as one way to accomplish this. Given the robustness of the EOQ cost function and the roughness of the input data, a reasonable approach to the multipart purchasing problem is to simply use the EOQ formula to compute an optimal order interval for each part (that is, D_j/Q_j^*) and then round to the nearest power of two of some convenient base ordering cycle. For instance, if weekly orders are practical, then round the EOQ interval to the nearest value in the set: 1 week, 2 weeks, 4 weeks, 8 weeks, etc.

To consider the overall capacity of the purchasing function, we could approach the problem as one of minimizing the total inventory holding cost for all parts subject to the constraint that the *average* order frequency not exceed some specified constant F. Since the total number of purchase orders placed per year is equal to the average order frequency per item multiplied by N, this formulation is equivalent to minimizing the total investment in inventory subject to the constraint that the total number of annual purchase orders not exceed NF. We have found it easier to think in terms of average order frequency, however, and therefore we state the problem in this way.

[4] Recall that in Part I we criticized the fixed-order-cost assumption for *production* systems because it frequently acts as a proxy for a capacity constraint, which changes over time and cannot be determined in advance of the schedule. However, for *purchasing* systems, capacity may not be a consideration, and therefore a fixed order cost is a much more plausible modeling assumption.

To formulate a mathematical model, we recall that if the order quantity for part j is Q_j, then the average inventory of part j (in units) is $Q_j/2$, and hence the annual holding cost is $h_i Q_i/2$. The order frequency of part j is D_j/Q_j. Therefore, total holding cost is $\sum_{j=1}^{N} h_j Q_j/2$, and the average order frequency is $1/N, \sum_{i=j}^{N} D_j/Q_j$. Thus, we can express the problem to minimize total holding cost subject to an average order frequency of no more than F as

$$\text{Minimum} \quad \frac{\sum_{j=1}^{N} h_j Q_j}{2} \tag{17.2}$$

$$\text{Subject to:} \quad \frac{1}{N}\sum_{j=1}^{N}\frac{D_j}{Q_j} \le F \tag{17.3}$$

Notice that if we replace holding cost h_j by unit cost c_j, then the problem becomes one of minimizing total inventory *investment* subject to a constraint on average order frequency. Some decision makers find it easier to think in terms of inventory investment rather than holding cost. However, the two are equivalent (i.e., result in the same lot sizes) if $h_j = ic_j$, where i is an interest rate. So the decision of whether to use holding cost or inventory investment as the objective is generally just a matter of taste.

This formulation is an example of a **nonlinear programming problem.** The standard technique for solving such problems is the **method of Lagrange,** which converts a constrained optimization problem to an unconstrained one by attaching a penalty to violation of the constraint and incorporating it into the objective (Bazaraa and Shetty 1979). While this sounds complex, it really boils down to finding a fixed setup cost for (17.1) that causes constraint (17.3) to be satisfied. We do this by an iterative search method like the following.

Algorithm (Multiproduct EOQ Model)

Step 0. Pick an initial value for A.

Step 1. Use A in Equation (17.1) to compute the lot sizes Q_j for all $j = 1, \ldots, N$.

Step 2. Compute the resulting order frequency:

$$F(A) = \frac{1}{N}\sum_{j=1}^{N}\frac{D_j}{Q_j}$$

Step 3. If $F(A) = F$, stop.[5] Else,
If $F(A) < F$, decrease A
If $F(A) > F$, increase A
and go to step 1.

The increases and decreases in A can be made by trial and error, or some more sophisticated search technique, such as interval bisection.[6] As long as the method we use takes smaller and smaller steps when we near the optimum, the procedure will eventually converge.

[5] Since $F(A)$ is a continuous number, it will never equal F exactly. So we typically stop when $F(A)$ is within some small prespecified tolerance of F.

[6] Basically, bisection starts with two points for A, an upper bound that is too high (i.e., causes $F(A) < F$) and a lower bound that is too low (i.e., causes $F(A) > F$), and tries the midpoint between them. If it is too high, then the midpoint replaces the upper bound; if it is too low, it replaces the lower bound. The gap between the lower and upper bounds will steadily decrease. When it is sufficiently small (i.e., below some specified tolerance), we stop.

At the end of this procedure, we will have the optimal order quantities Q_j^*, $j = 1, \ldots, N$. We also get the appropriate fixed order cost A. An alternate interpretation of this cost is the *decrease in total inventory holding cost per unit decrease in the average order frequency.* If we knew how much we were willing to pay in annual holding cost to decrease the average order frequency by one order per item per year, then we could immediately use this value in Equation (17.1) to compute the optimal order quantities. If, as is often the case, this is a difficult number to come up with, we can run the above algorithm for a variety of values of F and plot the optimal holding cost (or inventory investment, if we use c_j in place of h_j) versus average order frequency. Such a curve would represent the multiproduct analog to Figure 2.3 for the single-product case.

We could directly implement the optimal lot sizes Q_j, $j = 1, \ldots, N$, computed via the above procedure. However, if there are savings to ordering parts simultaneously, it may make sense to round the order intervals associated with these lot sizes to powers of two. We do this by noting that the reorder interval for part i is given by

$$T_j^* = \frac{Q_j^*}{D_j}$$

If we round the T_j^* values to the nearest power of two, then, as we discussed in Chapter 2, orders of different parts will tend to "line up." Of course, this rounding will affect both inventory and average order frequency. If we round the T_j^* values to T_j' values, then our order quantities become

$$Q_j' = T_j' D_j$$

Hence, the actual inventory holding cost will be

$$\frac{\sum_{i=j}^{N} c_j Q_j'}{2}$$

and the actual average order frequency will be

$$\frac{1}{N} \sum_{i=j}^{N} \frac{D_j}{Q_j'}$$

If the increase in inventory investment relative to the optimum is too great, or if the average order frequency is too much larger than the target level F, then the benefits from power-of-two rounding may not justify their costs. If the difference between the actual solution and the optimum is slight, then such rounding is probably worthwhile.

Example:

To illustrate the above procedure, we consider a very simple four-part example with data given in Table 17.1. The objective is to minimize average inventory investment subject to an average annual order frequency of $F = 12$ (i.e., once per month). Note that since the objective is average inventory *investment,* we use a holding cost rate equal to the unit cost $h_j = c_j$.

Table 17.2 summarizes the output of the above procedure applied to this example. The rightmost column in this table gives average inventory investment for each set of order quantities, which is calculated as

$$\frac{\sum_{i=j}^{N} c_j Q_j}{2}$$

To initiate the procedure, we begin with $A = 1$. As shown in Table 17.2, this results in an average order frequency of 96.85, which is much too high. Therefore, A must

TABLE 17.1 Input Data for Multipart Lot Size Example

Part j	D_j	c_j
1	1,000	100
2	1,000	10
3	100	100
4	100	10

TABLE 17.2 Calculations for Multipart Lot Size Example

Iteration	A	$Q_1(A)$	$Q_2(A)$	$Q_3(A)$	$Q_4(A)$	$F(A)$	Inventory Investment ($)
1	1.000	4.47	14.14	1.41	4.47	96.85	387.39
2	100.000	44.72	141.42	14.14	44.72	9.68	3,873.89
3	50.000	31.62	100.00	10.00	31.62	13.70	2,739.25
4	75.000	38.73	122.47	12.25	38.73	11.18	3,354.89
5	62.500	35.36	111.80	11.18	35.36	12.25	3,062.58
6	68.750	37.08	117.26	11.73	37.08	11.68	3,212.06
7	65.625	36.23	114.56	11.46	36.23	11.96	3,138.21
8	64.065	35.80	113.19	11.32	35.80	12.10	3,100.68
9	64.845	36.01	113.88	11.39	36.01	12.03	3,119.50
10	65.235	36.12	114.22	11.42	36.12	11.99	3,128.87
11	65.040	36.07	114.05	11.41	36.07	12.01	3,124.19
12	65.138	36.09	114.14	11.41	36.09	12.00	3,126.53

be increased. So we try $A = 100$. As we would expect, since we are penalizing frequent orders heavily, this results in much higher order quantities, and an average order frequency falls to 9.68. Since this is too low, we now have A bracketed. We know that the optimal value of A (the one that achieves an order frequency of 12) is between 1 and 100. So we try $A = 50$. Since this results in an order frequency of 13.70, it is too low. So we try $A = 75$. This decreases the order frequency to 11.18. Proceeding in this manner, the procedure eventually converges to the desired order frequency. Note that all the calculations involved are easily handled in a spreadsheet, provided that the number of parts is not too large. Indeed, it is a simple matter to use Goal Seek or Solver in Excel to search out the proper value of A.

The last line in Table 17.2 gives us the result from the multipart lot-sizing procedure. These numbers tell us that the optimal lot sizes for parts 1, 2, 3, and 4 are 36.09, 114.14, 11.41, and 36.09, respectively. Notice that the lot size of part 2 is larger than that of part 1, and the lot size of part 4 is larger than that of part 3. This is because part 2 is less costly than part 1 and part 4 is less costly than part 3. Intuitively, optimal lot size is decreasing in cost.

Furthermore, the lot size of part 1 is larger than the lot size of part 3, even though their costs are the same. This is because the demand is greater for part 1. The same

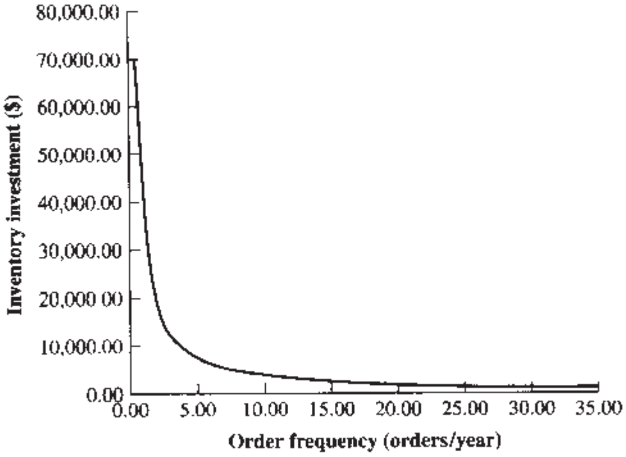


FIGURE 17.2

Inventory investment versus order frequency for multipart example

relationship holds between parts 2 and 4. As we would expect, lot size is increasing in demand rate.

Finally, notice that parts 1 and 4 have the same lot size. This is because

$$\frac{D_1}{c_1} = \frac{D_4}{c_4}$$

From expression (17.1), it is apparent that lot size depends on D_j and h_j (and hence c_j) only through their ratio.

The output from the procedure also tells us that $A = 65.138$. This gives us an estimate of the cost (in inventory investment) of changing the average order frequency. Increasing the order frequency by one (to 13 per year) would decrease inventory investment by \$65.14, while decreasing it by one (to 11 per year) would increase inventory investment by \$65.14. However, we must note that these costs are only approximate, since the true cost function is nonlinear. In reality, increasing the order frequency by one will save less than \$65.14, while decreasing it by one will cost more than \$65.14. However, it does give the user a rough idea of the inventory value of more frequent orders.

The resulting value of A also serves as a reality check on our original choice of order frequency target. If the actual cost of placing an order is less (more) than \$65.14, then we should have chosen an order frequency larger (smaller) than 12 times per year. The point is that if we have some idea of what A and F should be, but aren't completely certain about either, then we will get a better solution by cross-checking them against each other and adjusting until both are reasonable.

We can be more exact about the tradeoff between inventory investment and order frequency. Notice that if we keep track of the inventory investment, as we did in Table 17.2, then each choice of A gives us an inventory investment/order frequency pair. Hence, by varying A over a sufficiently wide range, we can generate a graph of inventory investment versus average order frequency. We do this in Figure 17.2. Notice that the inventory investment falls very rapidly as we increase the number of orders per year from zero to five. However, increasing the order frequency above this, and particularly above 10 per year, has a much smaller effect. This type of ***diminishing returns*** is exactly analogous to the behavior of the single-product model shown in Figure 2.3.

Last, if there are economies to joint orders, we might want to round our order intervals to powers of two. To do this, we first compute the order intervals:

$$T_1^* = \frac{Q_1^*}{D_1} = \frac{36.09}{1{,}000} = 0.03609 \text{ year} = 13.17 \text{ days}$$

$$T_2^* = \frac{Q_2^*}{D_2} = \frac{114.14}{1{,}000} = 0.11414 \text{ year} = 41.66 \text{ days}$$

$$T_3^* = \frac{Q_3^*}{D_3} = \frac{11.41}{100} = 0.11414 \text{ year} = 41.66 \text{ days}$$

$$T_4^* = \frac{Q_4^*}{D_4} = \frac{36.09}{100} = 0.3609 \text{ year} = 131.73 \text{ days}$$

Using days as our base time unit, we choose T_1' to be the closest power of two to 13.17, namely, $2^4 = 16$. We choose T_2' and T_3' as the closest power of two to 41.66, which is $2^5 = 32$. And we set T_4' equal to the closest power of two to 131.73, which is $2^7 = 128$. These order intervals translate to order quantities as follows:

$$Q_1' = \frac{D_1 T_1'}{365} = 1{,}000 \times \frac{16}{365} = 43.84 \text{ units}$$

$$Q_2' = \frac{D_2 T_2'}{365} = 1{,}000 \times \frac{32}{365} = 87.67 \text{ units}$$

$$Q_3' = \frac{D_3 T_3'}{365} = 100 \times \frac{32}{365} = 8.77 \text{ units}$$

$$Q_4' = \frac{D_4 T_4'}{365} = 100 \times \frac{128}{365} = 35.07 \text{ units}$$

Substituting these into the expressions for inventory investment and order frequency yields

$$\text{Inventory investment} = \frac{\sum_{j=1}^{4} c_j Q_j'}{2} = \$3{,}243.84$$

$$\text{Average order frequency} = \frac{1}{4}\sum_{j=1}^{4} \frac{D_j}{Q_j'} = 12.12$$

Since we presumably save some effort by combining orders due to the power of two order intervals, it may be acceptable to have a slightly higher average order frequency than the originally desired level of 12. Notice, however, that the inventory investment increases from \$3,126.53 to \$3,243.84. This increased cost must be offset by the benefits of joint replenishment (e.g., fewer separate purchase orders to issue, truck sharing) for the powers of two policy to be worthwhile.

17.4 Managing WIP

The first thing to note about managing WIP is that Little's law

$$\text{CT} = \frac{\text{WIP}}{\text{TH}}$$

implies that for fixed throughput, reducing WIP and reducing cycle time are directly linked. Therefore, the measures we will suggest to increase the efficiency of WIP are *precisely* the same as those one would use to reduce cycle times.

The second important point concerning WIP management is that, as we pointed out earlier, the bulk of work-in-process in most production systems (i.e., disconnected flow lines) is in queue (caused by variability and high utilization), waiting for batch (caused by batching), or waiting to match (caused by lack of synchronization). Thus, WIP reduction programs should be directed at (judiciously) lowering utilization, smoothing out variability, reducing batching, or improving synchronization.

In the following sections, we review techniques for reducing WIP in queue, waiting to move, and waiting to match.

17.4.1 Reducing Queueing

Recall that for a single-machine workstation, with mean processing time t_e, coefficient of variation of processing times c_e, coefficient of variation of arrivals c_a, and utilization u, cycle time can be approximated by

$$\mathrm{CT} \approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e + t_e \tag{17.4}$$

so by Little's law and the fact that $u = r_a t_e$, where r_a is the average arrival rate to the workstation,

$$\mathrm{WIP} = \mathrm{CT} \cdot r_a \approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)u + u \tag{17.5}$$

Thus, to reduce WIP and CT at the workstation, we can reduce the variability of arrivals to the station (c_a^2), the effective variability of the processing times at the station (c_e^2), or utilization (u).

Generic options for achieving these include the following:

1. **Equipment changes/additions.** The simplest way to increase capacity, and hence reduce utilization, of a station is to replace machines with faster models or augment the current machines with additional parallel capacity. While hardly imaginative, this option can be effective. However, to choose good equipment additions, we must consider the purchase cost, the effect on the capacity and variability at the station, and downstream (flow) variability effects. We discuss a framework for this in Chapter 18.

2. **Pull systems.** As we saw in Chapter 10, a pull system will achieve the same level of throughput with a lower average WIP level. The reason is that the releases to the line are coordinated with the status of the line (i.e., work is allowed to enter the line only when there is space for it). This is something like reducing c_a to the front of the line, but not quite. What pull systems really do is to tie releases to the line to completion of work within the line. Most importantly, they establish a WIP cap, which prevents the WIP level in a line from exceeding a specified quantity. Thus, pull systems can *mandate* a WIP reduction. The challenge is to achieve the WIP reduction without a loss in throughput. This requires making some of the other variability reduction or capacity enhancement changes suggested here.

3. **Finite-capacity scheduling.** If releases to the line are made without adequate attention to capacity (e.g., as in MRP), then WIP explosions at bottleneck resources are possible. As Chapter 15 described, a finite-capacity scheduling system can help regulate releases in accordance with system capacity. Although this does not tie releases to production quite as strongly as a pull system (a pull system links releases to *actual* production, while a finite-capacity scheduler links releases to *expected* production), finite-capacity schedulers can substantially reduce WIP by preventing systematic overreleasing to the line. Ideally, one should supplement a finite-capacity scheduling

system with a pull system, in order to keep the system under control when conditions depart from the schedule.

4. **Setup reduction.** All other things being equal, reducing setups will increase effective capacity, and therefore reduce utilization, of a workstation. However, typically when we reduce setups, we run smaller lots and hence perform more setups. Even if the increase in the number of setups completely offsets the capacity increase, as we discussed in Part II, shorter, more frequent setups will decrease effective variability at the workstation (c_e). This will serve to reduce queueing at the workstation and downstream (i.e., because flow variability will also be reduced). Moreover, as we noted earlier, if we can produce smaller batches, we will have less need to store excess production as finished goods inventory.

5. **Improved reliability/maintainability.** Increasing either the mean time to failure or the mean time to repair increases the availability of a machine and hence augments its capacity. In addition, decreasing the mean time to repair can significantly reduce the effective variability of the machine (c_e). Thus, these types of improvement can reduce queueing at a workstation and, by lowering downstream flow variability, also reduce queueing at subsequent stations.

6. **Enhanced quality.** As we noted in Chapter 12, reducing either rework or yield loss can substantially increase capacity and reduce effective variability. Because of this, quality improvement efforts can be major components of a WIP/cycle time reduction program.

7. **Floating work.** Cross-trained workers who can move to where capacity is required can increase the effective capacity of the line. Cross-training also tends to give workers a more global picture of the line and gets more brains thinking about the problems faced at each station in the line. In manual assembly systems, paced or unpaced, the effects of floating work can be achieved by designating certain tasks as "shared." For example, a particular component might be assigned to be attached by either worker A (upstream) or worker B (downstream). Whenever worker A is keeping up with the line, she will attach the shared component. However, if worker A gets behind (e.g., a quality glitch slows her down), then she can pass the component to worker B for him to attach. In general, floating work schemes only work effectively if the incentive system encourages cooperation toward a linewide goal (e.g., throughput).

Finally, we make the same point we made with regard to ABC classification of purchased parts: *Not all WIP need be treated equally.* It may make perfect sense to stratify parts by volume. High volume parts could be assigned to lines with few part families, and hence few setups, where the steadiness of flow facilitates use of a highly efficient pull system. Low volume parts could be produced in a job shop environment, so that high flexibility purchased at the cost of low efficiency would only affect a minor portion of the overall business. This type of **focused factory** strategy can greatly simplify management of a factory with many different parts.

17.4.2 Reducing Wait-for-Batch WIP

Batching for process reasons may be unavoidable (e.g., a batch burn-in operation that requires 24 hours may only be able to provide sufficient capacity when large batches are processed together). Batching for move reasons is another matter. Anything that enables jobs to move from one workstation to the next in smaller batches, and hence with less waiting, will clearly reduce WIP and cycle time. Specific approaches for doing this include these:

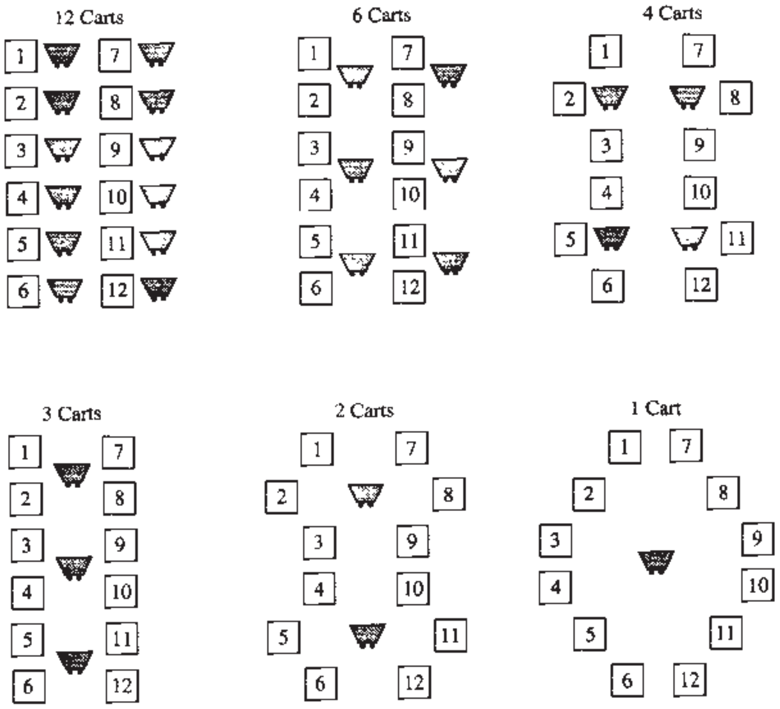
1. **Lot splitting.** Remember that **process lots** and **move lots** do not have to be the same. Even if long setup times at a workstation that processes jobs one at a time necessitate large batches for capacity reasons, there is no need to wait until the batch is complete before moving some of the jobs to the next workstation. For instance, a machining center that produces crankshafts in lots of 10,000 (i.e., before setting up to produce a different type of crankshaft) might send them to the subsequent finishing process in lots of 100. In theory, the crankshafts could even be moved one at a time from machining to finishing. The limiting factor is the amount of time required to move the material.

2. **Flow-oriented layout.** More frequent moves can be facilitated by the plant layout. One of the advantages of a cellular layout is that workstations are in close proximity so that material can move easily between them. Material handling systems (e.g., conveyors, AGVs) can also facilitate small lot transfer between workstations, even if they are not physically close to one another.

3. **Cart sharing.** In workstations with multiple parallel machines producing identical product, sharing incoming and/or outgoing carts (or whatever containers are used to move jobs between workstations) can reduce the amount of WIP waiting before and after the workstation. For instance, Figure 17.3 shows 12 machines filling different numbers of outgoing carts (we have not explicitly represented incoming carts). On average, the number of completed parts waiting to be moved to the next workstation in the system with one outgoing cart will be one-twelfth that in the system with 12 outgoing carts. Notice, however, that this assumes that the machine operators spend the same amount of time moving completed parts to the carts in both systems. If, because of geography, operators must walk farther to bring parts to the single shared cart than to put them

Figure 17.3

Cart-sharing arrangements



on individual carts in the 12-cart system, then cart sharing can lengthen the effective processing times. Depending on the system, the cycle time reduction from cart sharing might offset that from the capacity decrease. However, in general, cart sharing typically makes sense only where the time and inconvenience are slight. This consideration might make the three- or four-cart arrangement the most practical option for the 12-machine workstation in Figure 17.3.

17.4.3 Reducing Wait-to-Match WIP

At assembly stations, all subcomponents must be available in order for the assembly operation to occur. We have already discussed the problem of managing purchased parts feeding an assembly process in this chapter and in Chapter 12, so we will only consider the situation where subcomponents are produced on different fabrication lines within the plant.

Ideally, we would like to release work orders for the various subcomponents and process them in the fabrication lines so that they arrive at assembly at exactly the same time, in close coordination with the final assembly schedule. Variability generally makes this impossible, but there are things we can do to improve synchronization:

1. **Pull system.** As we know from Chapter 14, a pull system, and a CONWIP system in particular, will naturally synchronize releases into the fabrication lines with final assembly. If fabrication lines are of different length (i.e., in terms of the time required to traverse them), then different WIP levels (card counts) will be needed. This will mean that releases into the fabrication lines at the same time will not necessarily correspond to the same finished product. However, if the WIP levels in the fabrication lines are set appropriately, subcomponent arrivals to assembly will be synchronized.

2. **Common work backlog.** The above CONWIP scheme for coordinating releases with final assembly will only synchronize arrival of subcomponents to assembly if the release sequence is not scrambled in the fabrication lines. If, for instance, local dispatching rules such as shortest processing time (SPT) are used at individual workstations, then jobs can pass one another and synchronization will be lost. Even if we use first-in, first-out (FIFO) at the workstations in the fabrication lines, passing is still possible at multimachine stations. Thus, the way to maintain synchronization with the final assembly schedule is to follow a common **work backlog** at each workstation in the fabrication lines. This backlog simply lists the jobs in order of the final assembly sequence. As long as the fabrication workstations process jobs in the order specified by the backlog, the jobs will arrive synchronized to assembly. If the backlog must be routinely violated (e.g., because of batching or quality problems), then a buffer of WIP will have to be maintained in front of assembly to avoid stoppages due to "out-of-sync" arrivals.

3. **Balanced batching.** If one fabrication line uses large process lots because of a long setup, it may be unable to coordinate with the final assembly schedule. There are three ways to deal with this problem. (1) Produce well ahead of the final assembly schedule on this fabrication line, and maintain a substantial buffer between this line and final assembly. (2) Generate the final assembly schedule in accordance with the batching requirements of the fabrication line. (3) Reduce setup times or augment capacity in the fabrication line so that smaller lots become feasible and it can be synchronized with the desired final assembly schedule. The first two are short-term options; the third may require more time to implement.

17.5 Managing FGI

Finished goods inventory acts as a buffer between production and demand. As we noted earlier, such a buffer may be needed to (1) insulate customers from manufacturing cycle time, perhaps to provide "instant" delivery, (2) absorb variability in either the production or demand processes, or (3) level out capacity loading (e.g., due to seasonality). These imply that anything that links production and demand processes more closely will allow less FGI to be carried. Options for doing this include the following:

1. **Improved forecasting.** While we don't want to raise unrealistic expectations for a forecasting panacea, it is certainly the case that forecasting errors can inflate FGI. If better techniques for forecasting demand, like the time series methods of Chapter 13, can reduce the discrepancies between production and demand, then FGI will be reduced. Despite this fact, there are limits to our ability to predict the future, and so the other options below may be more promising in most systems.

2. **Dynamic lead time quoting.** Many systems quote fixed lead times to customers. However, because plant loading varies over time, actual manufacturing cycle times also vary over time. Therefore, if we set the fixed lead time such that the fraction of time we can deliver within this time is reasonably high, then a high percentage of jobs will finish early. If early delivery is not permitted, these jobs will wait in FGI. We can eliminate this problem by dynamically quoting customer lead times that are sensitive to plant loading.

For example, we worked with a manufacturer of metal cabinets that published 10-week fixed lead times in its product catalog. If it had used a dynamic lead time quoting system, customers who placed orders when the plant was almost empty might have received a two-week lead time, while customers who placed orders when the plant was backed up with work might have received a 12-week lead time. Overall, lead times would be shorter on average, and less product would have to wait in FGI for shipment to attain the same on-time delivery performance.

3. **Cycle time reduction.** A very effective way to reduce forecasting errors is to rely less on forecasting. If cycle time (including the entire value-added chain consisting of time to enter orders, code orders, engineer orders, schedule orders, manufacture products, deliver products, etc.) can be reduced, then work releases can be made closer to their due dates. Since forecasts tend to grow worse with distance into the future, later releases have the effect of making the master production schedule more reliable. If cycle times become short enough, then all releases can be made in conjunction with firm customer orders and therefore FGI due to forecasting errors can be eliminated altogether. Happily, all the WIP reduction techniques listed earlier are also cycle time reduction techniques (Little's law) and therefore are well suited to this purpose.

4. **Cycle time variability reduction.** Chapter 12 pointed out that if we want to guarantee a certain level of service, the lead time quoted to a customer is affected by both the average cycle time and the standard deviation of cycle time (see Figures 12.9 and 12.10). The more variability in cycle times, the more safety lead time we must build into our quotes to ensure a high percentage of on-time deliveries. Higher safety lead times imply that product will spend more time waiting in FGI, unless early delivery is permitted. Fortunately, many of the things we can do to reduce average cycle time (reduce setups, improve reliability/maintainability, implement pull mechanisms, reduce rework and scrap) also serve to reduce cycle time variance.

5. **Late customization.** Even if it is necessary to carry inventory in order to provide short customer lead times, it may not be necessary to carry the inventory in the form of FGI. In some cases, it may be possible to stock the product in semifinished form and assemble or customize to order. Semi-finished inventory is more flexible, provided it

can be used to produce more than one finished product, which makes it possible to carry less total inventory.

For example, a manufacturer of faucet fixtures might offer 20 different models made up of all combinations of five bases and four handle styles. By stocking the bases and handles, the manufacturer need maintain only nine different items in stock, instead of 20. Because of variability pooling, it is easier to forecast demand for the nine parts than for the 20 finished products, and hence less total stock will be required.

As another example, an appliance manufacturer might produce a family of electric mixers that differ according to accessories (a dough hook might or might not be included), retail outlet (labels and packaging might indicate a store brand), and market destination (instructions might be in different languages). By stocking generic families of mixers, distinguished by color of plastic parts, say, the manufacturer could quickly label and package mixers to supply demand for many different finished products. Under this strategy, forecasts would only have to be accurate at the family level, so FGI due to forecasting errors could be considerably reduced.

The potential drawbacks to this type of strategy are that (1) customer lead time is not reduced as much as if FGI is stocked in finished form, which could present a problem if the competition stocks at the FGI level, and (2) storage of semi-finished products can be difficult; for example, dirt and breakage might be a problem if mixers are not boxed.

The ability to store product at the semifinished level can also be a function of product design. For instance, the manufacturer of institutional cabinetry mentioned earlier had 10-week lead times in large part because of its large product line with each product built from scratch (i.e., sheet metal). A competitor was able to offer four-week lead times by offering a smaller product line built around a small set of standard modules (stocked) with different paint colors, face-framing options, and features (faucets, electrical hookups, glass doors, etc.) to allow them to meet customers' needs. Because customers were typically architects who were also frequently behind schedule, responsiveness was highly valued in this market, and the competitor was clearly gaining the upper hand as a result of the shrewd product design strategy.

6. **Balancing labor, capacity, and inventory.** In many markets, product is produced during periods of low demand and held as FGI to meet demand during peak periods. While this may be the best option in some cases, it is by no means the only way to address the problem of seasonal demand. An alternative approach may be to vary the size of the workforce, either by using temporary workers during the peak season or by pairing the product with one with an offset peak (e.g., lawnmowers with snowblowers) and transferring workers between lines. Another—heretical, to most traditional managers—option is to maintain enough excess capacity to meet peak demand without building inventory. When the costs of carrying FGI, obsolescence, and poor customer service due to forecasting errors are considered, it is possible that these other options may be more economical than building large stores of FGI. At the very least, it may make sense to use a combination of approaches, such as a limited inventory buildup, coupled with some excess capacity and some floating labor.

17.6 Managing Spare Parts

Managing spare parts is an important component of an overall maintenance policy, which can be a major determinant of operational efficiency in a manufacturing system. Because of its importance and complexity, a wide variety of spare parts practices are observed in industry (see Cohen, Zheng and Agrawal 1994 for a benchmark study). We will not attempt a survey of these practices. Instead, in this section, we establish a framework for evaluating spare parts inventories and build on the models from Chapter 2 to develop appropriate tools.

17.6.1 Stratifying Demand

There are two distinct types of spare parts, those used in scheduled **preventive maintenance** and those used in unscheduled **emergency repairs.** For instance, a filter may be used in a regular monthly maintenance procedure, while a fuse is replaced only when it fails. The two types of parts should be managed differently.

Scheduled maintenance represents a very predictable demand source. Indeed, if maintenance procedures are followed carefully, this demand may be much more stable than customer demand for finished products. Thus, standard MRP logic is probably applicable to these parts. That is, starting with projected demand, we net against current inventory (and scheduled receipts) and use a lot-sizing rule (lot for lot, fixed order quantity, etc.), to generate planned order receipts, and then back out according to purchasing lead times to generate purchase orders. If the parts are produced internally, we can substitute whatever scheduling procedure is used in place of the fixed purchase lead times to generate a production schedule. In either case, the stable predictable nature of the demand process makes these preventive maintenance parts relatively easy to manage.

Unscheduled emergency repairs are by definition unpredictable. Therefore, using MRP logic for these parts tends to work poorly. We address approaches for maintaining sufficient safety stock to support timely repair of equipment in the following section.

17.6.2 Stocking Spare Parts for Emergency Repairs

For spare parts whose demand is unpredictable, the challenge is to provide high service in a cost-efficient manner. Because demand is uncertain, the (Q, r) model we discussed in Chapter 2 is a potential tool for examining this tradeoff. To apply it, we must decide how to represent service in a multipart environment.

In spare parts systems, service is related to the availability of the machines being supported. Moreover, because a machine that is down for lack of a \$2 fuse is just as unavailable as one that is down for lack of a \$3,000 computer unit, it is often reasonable to assume that the cost of not having a part on hand is the same for all parts. Therefore, if we can specify either the backorder cost or the stockout cost, we can analyze the parts separately using one of the models of Section 2.4.3.

However, as we have noted before, backorder and stockout costs are often difficult to estimate. In the case of spare parts systems, the reason is that the cost of a part shortage depends on the cost of the machine outage caused by it, which in turn depends on the cost of customer delays caused by the outages. Because of this, it is frequently attractive to think of the problem in terms of a service constraint rather than a service cost. Fortunately, there is a close connection between the cost and constraint formulations.

To adapt the (Q, r) model to the multiproduct case, we make use of the same notation as in Section 2.4.3 with a subscript j to represent parameters for part j, $j = 1, \ldots, N$, so that

$$N = \text{total number of distinct part types in system}$$
$$D_j = \text{annual demand (units per year) for part } j$$
$$\ell_j = \text{replenishment lead time (days) for part } j$$

θ_j = expected demand during replenishment lead time for part j $(\theta_j = D_j \ell_j / 365)$

σ_j = standard deviation of demand during replenishment lead time for part j

$p_j(x)$ = probability of exactly x demands during replenishment lead time for part j (probability mass function)

$G_j(x)$ = $\sum_{y=0}^{x} p_j(y)$, probability that demand for part j during replenishment lead time is less than or equal to x (cumulative distribution function)

A = setup or purchase order cost per replenishment for any part (dollars)

c_j = unit production or purchase cost of part j (dollars per unit)

h_j = annual unit holding cost for part j (dollars per unit per year)

k = cost per stockout for any part (dollars)

b = annual unit backorder cost for any part (dollars per unit of backorder per year). Note that failure to have inventory available to fill a demand is penalized by using either k_j or b_j but not both.

B = desired total backorder level

S = desired average service level

F = desired average order frequency

Q_j = order quantity for part j (decision variable)

r_j = reorder point for part j (decision variable)

$F_j(Q_j)$ = order frequency (replenishment orders per year) for part j as a function of Q_j

$S_j(Q_j, r_j)$ = fill rate (fraction of orders filled from stock) of part j as a function of Q_j and r_j

$B_j(Q_j, r_j)$ = average number of outstanding backorders for part j as a function of Q_j and r_j

$I_j(Q_j, r_j)$ = average on-hand inventory level (units) of part j as a function of Q_j and r_j

With this notation, we can represent the total cost in two ways. We develop both, along with their associated constraint formulations, below.

Backorder Model. We begin by characterizing service by means of the average backorder level. We can formulate a cost function representing the sum of the setup plus backorder plus holding cost as

$$Y_b(\mathbf{Q}, \mathbf{r}) = \sum_{j=1}^{N} [AF_j(Q_j) + bB_j(Q_j, r_j) + h_j I_j(Q_j, r_j)] \tag{17.6}$$

where $\mathbf{Q} = (Q_j, j = 1, \ldots, N)$ and $\mathbf{r} = (r_j, j = 1, \ldots, N)$ represent vectors of the order quantities and reorder points. Since the cost function Y_b is simply the sum of separate terms that depend on (Q_j, r_j) pairs, we can minimize it by minimizing the terms for each j separately. But we already did this in Chapter 2. Hence, using the same approximation we used there (i.e., approximating the (Q, r) backorder formula $B_j(Q_j, r_j)$ by the base stock backorder formula $B_j(r_j)$) leads to the same expressions for the optimal order quantities and reorder points:

$$Q_j^* = \sqrt{\frac{2AD_j}{h_j}} \tag{17.7}$$

$$G(r_j^*) = \frac{b}{b+h_j} \tag{17.8}$$

Note that these are the familiar EOQ and base stock formulas. Furthermore, if we assume that lead time demand for product j is normally distributed with mean θ_j and standard deviation σ_j, then we can simplify (17.8) to

$$r_j^* = \theta_j + z_j\sigma_j \tag{17.9}$$

where z_j is the value in the standard normal table such that $\Phi(z_j) = b/(b + h_j)$.

Note that these expressions for Q_i and r_i are sensitive to the differences between parts. For instance, all other things being equal, a high-cost part (which will have a higher h_j coefficient) will have both a smaller order quantity Q_j and reorder point r_j than will a low-cost part. In addition, as we would expect, Q_j and r_j are increasing in the demand rate[7] D_j. In the normal demand case, the reorder point r_j will also increase in the standard deviation of lead time demand provided that $z_j > 0$, which as we noted in Chapter 2 is true as long as $b > h_j$. Finally, we note that increasing the fixed order cost A increases all order quantities Q_j, and increasing the backorder cost b increases all reorder points r_j.

If we can specify reasonable values for the fixed setup (order) cost A and the unit backorder penalty b, we can use formulas (17.7) and (17.9) to compute stocking parameters for the multiproduct (Q, r) system. However, as we observed in Chapter 2, this is frequently difficult to do in practice. In production environments, A is often a proxy for capacity, since the motivation for producing in batches is to avoid capacity losses due to frequent setups. In purchasing environments where capacity is not a direct concern, estimating A directly is much easier. But even in this case, estimating the backorder cost b is problematic, since it involves placing a value on loss of customer goodwill and other intangibles. For this reason, it is often more intuitive to use a constrained model. When service is appropriately characterized by the total number of outstanding backorders (for all part types), then we can formulate the problem as:

Minimize	Inventory holding cost
Subject to:	Average order frequency $\leq F$ Total backorder level $\leq B$

We can use an iterative procedure, like that we described for the multiproduct EOQ model earlier, to solve this constrained problem. The basic idea is to first adjust the fixed order cost A until the order frequency constraint is satisfied and then adjust the backorder cost b until the backorder level constraint is satisfied. Note that when we check to see whether a given set of (Q_j, r_j) values satisfies the backorder level constraint, we use the *exact* formula for computing backorder level, not the approximation we used to derive Equation (17.8). Also, because the backorder level $B_j(Q_j, r_j)$ depends on both Q_j and r_j, while the order frequency $F_j(Q_j) = D_j/Q_j$ depends only on Q_j, it is important to adjust A first and b second. We state the procedure formally on the next page.

[7] To see that r_j increases in D_j, note that increasing D_j increases θ_j and by Equation (17.9) we see that r_j increases in θ_j.

Algorithm (Multiproduct (Q, r) Backorder Model)

Step 0. Pick initial values for A and b.

Step 1. Use A in Equation (17.7) to compute the lot sizes Q_j for all $j = 1, \ldots, N$.

Step 2. Compute the resulting order frequency

$$F(A) = \frac{1}{N}\sum_{j=1}^{N}\frac{D_j}{Q_j}$$

Step 3. If $F(A) = F$, go to Step 4. Else,
If $F(A) < F$, decrease A
If $F(A) > F$, increase A
and go to step 1.

Step 4. Use b in Equation (17.9) to compute the reorder points r_j for all $j = 1, \ldots, N$.

Step 5. Compute the resulting total backorder level

$$B(b) = \sum_{j=1}^{N} B_j(Q_j, r_j)$$

Step 6. If $B(b) = B$, stop. Else,
If $B(b) < B$, decrease b
If $B(b) > B$, increase b
and go to step 4.

Stockout Model. If service is characterized better by average fill rate than by total backorder level, then we can formulate a cost function representing the sum of the setup plus stockout plus holding cost as

$$Y_s(\mathbf{Q}, \mathbf{r}) = \sum_{j=1}^{N}\left\{AF_j(Q_j) + k[1 - S_j(Q_j, r_j)] + h_j I_j(Q_j, r_j)\right\} \tag{17.10}$$

where $\mathbf{Q} = (Q_j, j = 1, \ldots, N)$ and $\mathbf{r} = (r_j, j = 1, \ldots, N)$ represent vectors of the order quantities and reorder points. As with the backorder cost model, we can optimize this separately for each part j. Using the same approximation we used in Chapter 2 (i.e., that we can compute Q_j using the EOQ model and approximate the fill rate with the type II approximation $S_j(Q_j, r_j) \approx 1 - B_j(r_j)/Q_j$ and approximate the backorder level $B_j(Q_j, r_j)$ by the base stock backorder formula $B_j(r_j)$) leads to the same expressions for the optimal order quantities and reorder points:

$$Q_j^* = \sqrt{\frac{2AD_j}{h_j}} \tag{17.11}$$

$$G(r_j^*) = \frac{kD_j}{kD_j + h_j Q_j} \tag{17.12}$$

If we further assume that lead time demand for product j is normally distributed with mean θ_j and standard deviation σ_j, then we can simplify Equation (17.12) to

$$r_j^* = \theta_j + z_j\sigma_j \tag{17.13}$$

where z_j is the value in the standard normal table such that $\Phi(z_j) = kD_j/(kD_j + h_j Q_j)$.

As in the backorder model, these expressions for Q_i and r_i are sensitive to the differences between parts. Again, all other things being equal, a high-cost part will have both a smaller order quantity Q_j and reorder point r_j than will a low-cost part. Also, Q_j and r_j are again increasing in the demand rate D_j, and in the normal demand case, the reorder point r_j will increase in the standard deviation of lead time demand provided that $z_j > 0$. Finally, as we would expect, increasing the fixed order cost A increases all order quantities Q_j, and increasing the stockout cost k increases all reorder points r_j. A difference from the backorder model is that the r_j^* values depend on the Q_j values.

If we can specify reasonable values for the fixed setup (order) cost A and the unit stockout penalty k, we can use formulas (17.11) and (17.13) to compute stocking parameters for the multiproduct (Q, r) system. If, for the reasons discussed and in Chapter 2, we are not able to do this, we can use a constrained formulation. When service is appropriately characterized by the *average* fill rate, then we can formulate the problem as

Minimize	Inventory holding cost
Subject to:	Average order frequency $\leq F$
	Average fill rate $\geq S$

We can use an analogous iterative procedure to that used above for the backorder model. As before, we make use of *exact* formulas for computing the fill rate in order to check the fill rate constraint. Again, it is important to adjust A to achieve the order frequency constraint before adjusting k to achieve the fill rate constraint. The formal procedure can be stated as follows:

Algorithm (Multiproduct (Q, r) Stockout Model)

Step 0. Pick initial values for A and k.

Step 1. Use A in Equation (17.11) to compute the lot sizes Q_j for all $j = 1, \ldots, N$.

Step 2. Compute the resulting order frequency

$$F(A) = \frac{1}{N} \sum_{j=1}^{N} \frac{D_j}{Q_j}$$

Step 3. If $F(A) = F$, go to step 4. Else,
    If $F(A) < F$, decrease A
    If $F(A) > F$, increase A
  and go to step 1.

Step 4. Use k in Equation (17.13) to compute the reorder points r_j for all $j = 1, \ldots, N$.

Step 5. Compute the resulting total average fill rate

$$S(k) = \frac{\sum_{j=1}^{N} D_j S_j(Q_j, r_j)}{\sum_{j=1}^{N} D_j}$$

Step 6. If $S(k) = S$, stop. Else,
    If $S(k) < S$, increase k
    If $S(k) > S$, decrease k
  and go to step 4.

Multiproduct (Q, r) Example. To illustrate the use of the backorder and stockout models for the multiproduct (Q, r) problem, and the difference between them, we consider the example in Table 17.3. This table gives the unit cost c_j, annual demand D_j,

Table 17.3 Cost and Demand Data for Multipart (Q, r) Example

j	c_j (\$/unit)	D_j (units/yr)	ℓ_j (days)	θ_j (units)	σ_j (units)
1	100	1,000	60	164.4	12.8
2	10	1,000	30	82.2	9.1
3	100	100	100	27.4	5.2
4	10	100	15	4.1	2.0

Table 17.4 Results of Multipart Stockout Model (Q, r) Calculations

j	Q_j (units)	$kD_j/(kD_j + h_jQ_j)$ (unitless)	r_j (units)	F_j (Order Freq.)	S_j (Fill Rate)	B_j (Backorder Level)	I_j (Inventory Level)(\$)
1	36.1	0.666	169.9	27.7	0.922	0.544	2,410.66
2	114.1	0.863	92.1	8.8	0.995	0.022	670.24
3	11.4	0.387	25.9	8.8	0.749	0.918	512.52
4	36.1	0.666	5.0	2.8	0.988	0.014	189.33
				12.0	0.950	1.497	3,782.75

replenishment lead time ℓ_j, and mean and standard deviation of lead time demand, θ_i and σ_i, respectively. Our objective is to minimize average inventory investment subject to constraints on average order frequency and either average fill rate or average backorder level. Note that since we are using inventory investment as our objective, we set the holding cost equal to unit cost: $h_j = c_j$.

First we address the problem of setting the order quantities Q_j. To do this, we assume a target average order frequency of $F = 12$ orders per year. Notice that the unit cost and annual demand data are identical to those in Table 17.1. Hence, we have already solved this problem because the portion of the multipart algorithms for computing Q_j is identical to the multipart EOQ algorithm. From our previous example, we know that choosing a fixed order cost of $A = 65.138$ yields Q_j values that achieve an average order frequency of 12 per year. These Q_j values are recorded in Tables 17.4 and 17.5.

This leaves only the problem of computing the reorder points r_j. We start by using the stockout model with a target average fill rate of $S = 0.95$. Using the above stockout model algorithm, we find that the penalty cost that makes the average fill rate equal 95 percent is $k = 7.213$. Table 17.4 reports the resulting critical ratios, reorder points, fill rates, backorder levels, and inventory levels for each part. It also computes the average fill rate (95 percent), the total backorder level (1.497 units), and the total inventory investment (\$3,782.75).

Notice that the algorithm produces a very high fill rate (99.5 percent) for inexpensive, high-demand part 2, but a low fill rate (74.9 percent) for expensive, low-demand part 3. Intuitively, the algorithm is trying to achieve an average fill rate of 95 percent as cheaply as possible, so it makes service high where it can do so cheaply (i.e., where the unit cost is low) and where it has a big impact on the overall average (i.e., where annual demand is high).

TABLE 17.5 Results of Multipart Backorder Model (Q, r) Calculations

j	$b_j/(b_j+h)$ (unitless)	Q_j (units)	r_j (units)	F_j (Order Freq.)	S_j (Fill Rate)	B_j (Backorder Level)	I_j (Inventory Level)($)
1	0.538	36.1	165.6	27.7	0.875	0.974	2,024.77
2	0.921	114.1	95.0	8.8	0.997	0.010	698.76
3	0.538	11.4	27.9	8.8	0.840	0.511	671.85
4	0.921	36.1	7.0	2.8	0.998	0.002	209.10
				12.0	0.934	1.497	3,604.48

An alternative to characterizing service via fill rate is to use the backorder level instead. We can do this by using the backorder model algorithm to adjust the backorder cost b until the total backorder level achieves a specified target. To make a comparison of the stockout and backorder models, we take as our total backorder target the level that resulted from the stockout model, that is, $B = 1.497$ units.

Before going on, we pause to note that establishing a target backorder level is not always an easy thing to do. Unlike the fill rate, which is expressed in a unitless percentage, the total backorder level measures the average number of outstanding backorders at any time. Therefore, one cannot easily translate a backorder level from one system to another (e.g., an average backorder level of five might be horrendous service for a system with few parts and low demand and just fine for a system with many parts and high demand). One way to place the backorder level in a more intuitive context is to think of it in terms of the average wait a customer demand experiences as a result of backorders. If we let W represent the average wait of a demand and D represent the total number of demands per year, then by Little's law

$$B = D \times W$$

or

$$W = \frac{B}{D}$$

In this example, $D = 2{,}200$ units per year, so a backorder level of 1.497 units translates to

$$W = \frac{1.497}{2{,}200} = 6.8045 \times 10^{-4} \text{ years} = 5.96 \text{ hours}$$

This means that on average a part (any part, not just one that encounters a backorder situation) will experience 5.96 hours of delay due to lack of inventory. Of course, what this really means is that most parts will encounter no delay, while others will experience significantly longer than 5.96 hours. But looking at the average delay per part gives the decision maker a sense of how much disruption is implied by a given backorder level. Indeed, it is completely equivalent to use hours of delay as the performance target instead of backorder level in the algorithm—all we have to do is to divide by the demand rate and multiply by the number of hours in a year.

Now, supposing that the backorder level target of 1.497 is reasonable, we can use the backorder algorithm to find the backorder penalty that causes total backorders to achieve this level. It turns out that $b = 116.50$ does the trick. Table 17.5 reports the resulting critical ratios, reorder points, fill rates, backorder levels, and inventory levels for each part. It also computes the average fill rate (93.4 percent), the total backorder level (1.497 units), and the total inventory investment ($3,604.48).

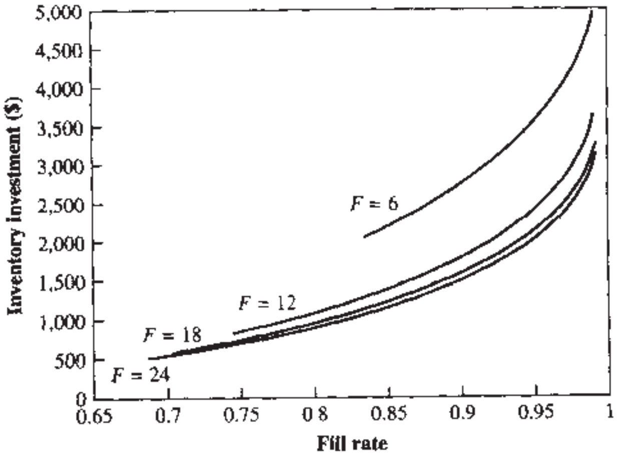
Notice that the algorithm results in low backorder levels for inexpensive parts 2 and 4, but higher backorder levels for expensive parts 1 and 3. In addition, it tends to have higher backorder levels for higher-demand parts (i.e., part 1 is higher than part 3, and part 2 is higher than part 4) because higher demand produces more backorders when all other things are equal. As did the stockout model, the backorder model places the bulk of its inventory investment in the expensive, high-demand part 1.

But there are some key differences between the two solutions. Notice that while the total backorder levels are the same, as we forced them to be, the fill rates and inventory levels are different. The backorder model achieves a given backorder level with a smaller investment in inventory ($3,604.48 versus $3,782.75). But it does so at the price of a lower fill rate (93.4 percent versus 95 percent). If we had used the backorder model to adjust the backorder cost b to make the fill rate equal 95 percent, it would have resulted in a higher inventory investment than did the stockout model. The conclusion is that the stockout model finds a policy that efficiently uses inventory to achieve a given fill rate, while the backorder model finds a policy that efficiently uses inventory to achieve a given total backorder level. Thankfully, this is exactly what we would expect them to do. But since the two models articulate different tradeoffs, it is important that we choose the right one for a given situation. If fill rate is the right measure of service, the stockout model is appropriate. If backorder level (or time delay) is a better representation of service, then the backorder model makes more sense.

Finally, we observe that we can use either the stockout or the base stock model to generate a tradeoff curve between inventory investment and either fill rate or backorder level. We do this by simply varying the stockout cost k or the backorder cost b and plotting the resulting pairs of inventory investment and fill rate (or backorder level). Figure 17.4 depicts curves for the previous example for a variety of order frequencies. Note that, as we expect, inventory investment grows exponentially as we approach a 100 percent fill rate. Furthermore, we can see that the inventory reduction from adding an additional six replenishment orders per year diminishes as the number of orders increases. These curves represent **efficient frontiers,** since they represent the lowest inventory investment for each order frequency/fill rate pair. A manager can use a graph like this

Figure 17.4

Tradeoff between order frequency, fill rate, and inventory investment in multipart (Q, r) model



to get a feel for how much investment in inventory is required to achieve various service levels. With this information, he or she can choose a sensible fill rate target. A similar curve of inventory investment versus fill rate could be generated by using the backorder model.

17.7 Multiechelon Supply Chains

Many supply chains, including those for spare parts, involve multiple levels as well as multiple parts. For instance, a retailer might stock inventory in regional warehouses, which supply individual outlets, which in turn supply customers. Alternatively, an equipment manufacturer that offers service contracts on its products may stock spare parts in a main distribution center, which supplies regional facilities, which in turn provide parts to maintain customer equipment. Because of variability pooling, stocking inventory in a central location, such as a warehouse or distribution center, allows holding less safety stock than holding separate inventories at individual demand sites. However, holding inventory in distributed fashion (e.g., at the retail outlets or service facilities) enables swifter response to demand because of geographic proximity. The basic challenge in multiechelon supply chains is to balance the efficiency of central inventories with the responsiveness of distributed inventories so as to provide high system performance without excessive investment in inventory. Research indicates that doing this by directly applying single-level approaches to multilevel problems can work poorly (Hausman and Erkip 1994, Muckstadt and Thomas 1980). This motivates us to give multiechelon systems special treatment.

The complexity and variety of multiechelon supply chains make them very challenging from an analysis standpoint. Serious study of such systems dates back to the classical work of Clark and Scarf (1960) and continues today (see Federgruen 1993, Axsäter 1993, Nahmias and Smith 1992 for excellent surveys and Schwartz 1981 for an anthology on the subject). More modern studies place multiechelon inventory management in the context of supply chain management (see, e.g., Lee and Billington 1992; Fisher 1997; Simchi-Levi, Kaminsky, and Simchi-Levi 1999). Since it is not possible for us to give anything close to a comprehensive treatment here, we will focus instead on defining the issues and indicating how some of the earlier single-level results can be adapted to the multilevel setting.

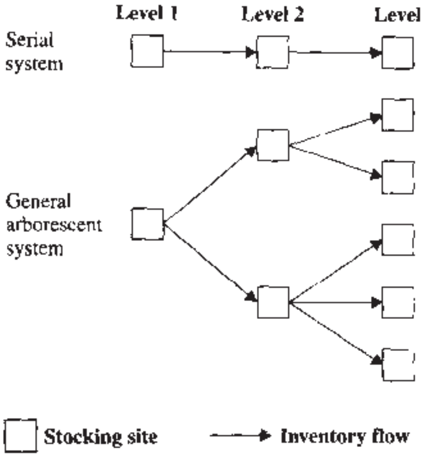
17.7.1 System Configurations

The defining feature of a multiechelon supply chain is that lower-level locations are supplied by higher-level locations. However, within this framework there are many possible variations, and, if we allow transshipment between locations at the same level (e.g., regional warehouses can supply one another), then the very definition of a level becomes hazy. In short, multiechelon systems can be extremely complex.

For the purposes of our discussion, we will concentrate primarily on **arborescent** systems, in which each inventory location is supplied by a single source (see Figure 17.5). In particular, we will consider the two-level arborescent system in which a single central warehouse (depot, distribution center) supplies multiple retail outlets (facilities, demand sites). We do this because (1) such systems are common in practice; (2) good approximate models of their behavior exist (see Deuermeyer and Schwartz 1981, Sherbrooke 1992, Svoronos and Zipkin 1988); and (3) approaches to the two-level problem can be used as building blocks for developing approaches to more complex multilevel systems.

FIGURE 17.5

Arborescent multiechelon supply chains



Before we move on to analysis, however, it is important to point out that the system configuration itself is a decision variable. Just because a system is currently configured using a three-level arborescent structure does not mean that this must always be the case. Indeed, determining the number of inventory levels, the locations of warehouses, and the policies for interconnecting them can be among the most important logistics decisions a firm can make about its distribution system. Even though these systems present challenging problems, it is better to address them openly than to miss significant opportunities because the status quo is viewed as immovable.

As an example of this type of rethinking the system configuration, we offer the case of an equipment manufacturer with whom we are familiar. This firm offered service contracts on its equipment (e.g., a guarantee of a maximum number of hours of downtime per month) and stocked spare parts to support the maintenance process. These parts were stocked at three levels: at a main distribution center, at regional facilities, and at customer sites (for customers whose service contracts specified it). Virtually all shipments from the distribution center to facilities were made via overnight mail (except for one facility that was close enough to the distribution center for the maintenance personnel to physically pick up parts needed for repairs). Maintenance personnel replenished on-site inventories from the facilities. Roughly one-half of the total inventory in the system was held at the distribution center, with the remainder in the field (i.e., at facilities and sites).

This configuration raises an obvious question. Why stock parts at a distribution center at all?[8] A facility can receive a part overnight equally well from the distribution center or from another facility. (Indeed, we discovered that the facility managers had an informal system of getting parts from one another via overnight mail when the distribution center was out of stock.) Thus, it might be possible for the distribution center to divide its inventories among the facilities. This would place the inventory geographically closer to the demand sites and therefore make it less likely that customers with broken machines would have to wait overnight for a crucial part. Moreover, if a facility lacked a part, it could still get it overnight, from another facility instead of the distribution center, provided that some facility in the system had the part in stock. The distribution center would cease to be a physical stocking site and would become the logical purchasing agent (i.e., to order parts from vendors or to be manufactured internally) and coordinating

[8] We are indebted to Professor Yehuda Bassok for pointing out this "obvious" question to us.

mechanism (i.e., by maintaining the information system that kept track of the location of the inventory in the system). The net result would be that for the same total amount of inventory in the system, customers would receive better repair service. This kind of bold reconfiguration might well offer greater overall benefits than detailed optimization of the existing system.

17.7.2 Performance Measures

To make design decisions or develop a model, it is essential that desired system performance be specified in concrete terms. A host of measures can be used, including these:

1. **Fill rate** is the fraction of demands that are met out of stock. This could apply at any level in the system. It is important to remember, however, that a measure applied to higher levels (e.g., the central warehouse) is only a means to an end. It is the performance of the low levels that actually service customers that determines the ultimate performance of the system.

2. **Backorder level** is the average number of orders waiting to be filled. This measure applies to systems where backordering occurs (e.g., spare parts systems, where a demand must eventually be filled whether or not the part is in stock at the time of the demand). As we noted earlier, backorder level is closely related to the average backorder delay, since we can apply Little's law to conclude

$$\text{Average backorder delay} = \frac{\text{Average backorder level}}{\text{Average demand rate}}$$

For instance, if a particular part has an annual usage of 100 parts per year and the average backorder level is one part, then the average delay seen by a part (any part, not just those that get backordered) is $\frac{1}{100}$ year, or 3.65 days.

3. **Lost sales** is the number of potential orders lost due to stockout. This measure applies to systems in which customers go elsewhere rather than wait for a backordered item (e.g., retail outlets). If every demand that encounters a stockout situation is lost, then the expected lost sales per year is related to fill rate by

$$\text{Lost sales} = (1 - \text{Fill rate}) \times \text{Average demand rate}$$

For instance, if the fill rate for a given part is 95 percent and annual demand is 100 parts per year, then $(1 - 0.95)(100) = 5$ parts per year will be lost due to stockout.

4. **Probability of delay** is the likelihood that an activity (e.g., a machine repair, shipment of a multipart customer order) will be delayed for lack of inventory. This measure is often used in systems where high reliability is required (e.g., aircraft maintenance). In general, the probability of delay in a multipart, multilevel system is a function of the fill rates of the various parts, although depending on the manner in which parts are demanded together (e.g., used on the same repair or customer order), this dependence can be complex (see Sherbrooke 1992 for a more complete discussion).

From these discussions we conclude that fill rate and average backorder level are key measures, since the other measures can be computed from them. For this reason, the majority of mathematical models either use these measures directly or use cost functions that rely on them.

17.7.3 The Bullwhip Effect

An important issue that arises in multiechelon supply chains is that of **channel alignment.** This refers to the coordination of policies between the various levels and can involve information sharing, inventory control, and transportation, among other management decisions. Because there are so many possible decision variables, channel coordination is challenging even when a single firm controls all the levels in the supply chain. When the levels consist of different firms, the problem becomes even more daunting.

A natural response to the complexity of multiechelon supply chains is to treat the various levels independently. That is, allow each level to use local information to implement locally "optimal" policies. Indeed, when levels consist of separate firms, such a strategy is the traditional default. But while natural to implement, the approach of separating levels can lead to very poor performance of the overall supply chain. The most obvious consequence of poor channel coordination is inefficiency (i.e., inventory will be held in inefficient quantities and locations). But a more subtle, though equally damaging, consequence is the **bullwhip effect,** which refers to the amplification of demand fluctuations from the bottom of the supply chain to the top.

Figure 17.6 illustrates the bullwhip effect. Even though demand at the bottom of the supply chain (e.g., retail level) is relatively stable over time, it is quite volatile at the top level (e.g., manufacturer level). This phenomenon was observed by Forrester (1961) in case studies of industrial dynamics models. It was also noted in a behavioral context as part of the well-known Beer Game, developed at MIT in the 1960s (see Sterman 1989). More recently it has been observed in practice. For example, Procter & Gamble noted that retail demand for Pamper brand diapers was fairly stable, while distributor orders to the manufacturing plant were highly variable. Similar behavior has been observed in the demand for printers by Hewlett-Packard and for insulin produced by Eli Lilly. As we know, variability must be buffered—by inventory, capacity, or time. Hence, the bullwhip effect leads to negative consequences, such as excessive WIP, poor use of capacity, long customer backlogs, and expediting costs.

Given that the bullwhip effect is real, the key questions are, What causes it? and What can be done about it? Lee, Padmanabhan, and Whang (1997a, 1997b) classified the causes of the bullwhip effect into four categories. Following their structure, we will summarize these along with potential remedies.

Batching. At the lowest level of the supply chain (e.g, the retail level) demand is often steady, or at least predictable, because purchases are made in small quantities.

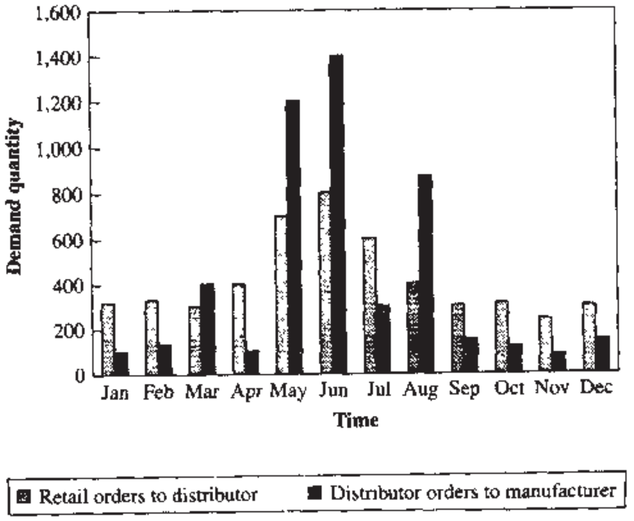


FIGURE 17.6

Demand seen by different levels of the supply chain

For instance, individual diabetics typically purchase small supplies of insulin, adequate to meet needs for a few weeks or months. Since the diabetics make their decisions independently, total retail demand is extremely level over time. This smoothness would be preserved throughout the supply chain if the retailer replenished its stock directly by placing lot-for-lot orders on the distributor, and the distributor did the same with its orders to the manufacturer. However, if retailers and distributors use some kind of lot-sizing rule (e.g., they follow a (Q, r) policy and hence wait until their requirements justify a replenishment order of size Q), then their demands will be much lumpier than those at the retail level. Furthermore, if there is synchronization among the decision makers at a given level (e.g., they all regenerate their MRP systems at the beginning of the month[9]), then this lumpiness will be even more exaggerated.

Since the amplification of demand variability is the result of batch ordering, policies that facilitate replenishment of stock in smaller quantities will reduce the bullwhip effect. Some options are to

1. *Reduce the cost of the replenishment order.* As we know from Chapter 2, one of the main reasons for ordering in bulk is the cost of placing a purchase order. One way to lower this cost is by using **electronic data interchange (EDI)** to reduce or eliminate purchase orders. By greatly reducing the amount of paperwork involved, such "paperless" ordering systems can facilitate more frequent replenishment in smaller quantities.

2. *Consolidate the orders to fill the trucks.* Another reason for ordering in bulk is the cost of transportation. It is not uncommon for wholesalers or distributors to set their order quantities equal to a full truckload. This is because the cost of shipping in full truckloads is significantly less than that for less-than-full truckloads. However, a truck need not necessarily be filled with the same product. So one way to reduce order quantities while retaining the full-truck cost advantage is to order multiple products from the same supplier. Alternatively, the replenishment process could be turned over to a third-party logistics company, which would consolidate loads from multiple suppliers and/or multiple customers. In either case, the result would facilitate more frequent deliveries.

Forecasting. In supply chains where the levels are managed by independent decision makers (e.g., they consist of separate companies), demand forecasting can amplify order variability. To see how, suppose that the retailer sees a small spike in demand. Because orders must cover both anticipated demand and safety stock, this leads to an order spike that is larger than the demand spike. The distributor, who forecasts demand on the basis of retailer orders, sees this spike, adds its own safety stock to the anticipated demand, and passes on an even larger order spike to the manufacturer. The reverse situation happens when the retailer sees a dip in demand. Hence, demand volatility increases as we progress up the supply chain.

The basic reason that forecasting aggravates the bullwhip effect is that each level updates its forecast on the basis of the demand *it sees,* rather than on actual customer demand. Hence, policies that serve to consolidate demand forecasting will reduce the bullwhip effect. Some options include these:

1. *Share demand data.* A simple remedy for reducing the amplification effect of separate forecasting at multiple levels is to use a common set of demand data. In supply chains owned by a single firm, sharing demand data from the lowest level is conceptually

[9] The phenomenon of synchronized MRP systems causing total demand to spike at certain times is sometimes called the **MRP jitters.**

straightforward (although far from universally practiced). In supply chains involving multiple firms, it requires explicit cooperation. For example, IBM, Hewlett-Packard, and Apple all require sell-through data from their resellers as part of their contracts. In supply chains where the participants make use of EDI, information sharing is relatively simple in principle; the challenge is to achieve the necessary degree of partnering to make it happen.

2. *Vendor-managed inventory.* A more aggressive way to ensure that forecasting is done using low-level demand data is to have a single entity do it. In **vendor-managed inventory (VMI)** systems, the manufacturer controls resupply over the entire chain. For example, Proctor & Gamble controls inventories of Pampers all the way from its supplier (3M) to its customer (Wal-Mart). The fact that alliances using VMI can pool inventory across levels enables them to operate with substantially less inventory than is needed in uncoordinated supply chains.

3. *Lead time reduction.* The magnification effect of forecasting on orders is a function of the amount of safety stock a demand spike drives into the system. But as we saw in Chapter 2, safety stock increases with replenishment lead time. Hence, an obvious, but potentially significant, way to reduce demand volatility due to forecasting is through lead time reduction. Any of the efficiency improvements discussed in Section 17.4 for WIP/cycle time reduction could be practiced at the various levels to achieve this.

Pricing. Another factor that can cause demand seen at higher levels of the supply chain to "clump up" into spikes is price discounting. Whenever a product's price is low, due to promotional pricing, customers tend to forward-buy (i.e., purchase in greater quantities than needed). When prices return to normal, customers consume the excess stock and hence order less than normal. The result is a volatile demand process.

Since it is price variation that drives demand volatility, the obvious remedy is to stabilize prices. Specific policies for supporting more stable prices are

1. *Everyday low pricing.* The most straightforward way to stabilize prices is to simply reduce or eliminate reliance on promotions using discounting. In the grocery industry, several manufacturers have established uniform wholesale pricing policies and have promoted them via a marketing campaign centered on "everyday low prices" or "value prices."

2. *Activity-based costing.* Traditional accounting systems may not show the costs of some practices resulting from promotional pricing, such as when regional discounts cause retailers to buy in bulk in one area and ship product to other areas for consumption. Activity-based costing (ABC) systems account for inventory, shipping, handling, etc., and hence are useful in justifying and implementing an everyday low-pricing strategy.

Gaming Behavior. One final factor that contributes to the bullwhip effect is the manner in which customers use their orders in a gaming fashion. For instance, suppose a supplier allocates a product in short supply to customers in proportion to the quantities they have on order. Then customers have a clear incentive to exaggerate their orders in hope of getting more product. When supply catches up with demand, the customers will cancel the excess orders, leaving the supplier awash in inventory. This occurred more than once during the 1980s in the computer memory chip market, when shortages encouraged computer makers to order chips from several suppliers, buy from the first one to deliver, and cancel the remaining orders.

The fundamental issue here is that when gaming behavior is present, customer orders can provide very bad information to the supplier about actual demand. Alternatives for reducing the incentive to game orders include the following:

1. *Allocate shortages according to past sales.* If a supplier facing a product shortage allocates its supply on the basis of historical demand, rather than current orders, then customers do not have an incentive to exaggerate orders in shortage situations.

2. *Use more stringent time fencing.* Recall from Chapter 3 that frozen zones and time fences are tools used to place restrictions or penalties on customers for making changes in orders. If customers cannot freely cancel orders, then gaming strategies become more costly. Of course, a supplier must decide on a reasonable balance between responsive customer service and demand stabilization.

3. *Reduce lead time.* Another situation that can lead to gaming behavior occurs when products involve long-lead-time components. For example, we worked with a printed-circuit board (PCB) plant that supplied computer assembly (box) plants. To assemble the circuit boards, the PCB plant had to purchase both the raw cards and the components to be mounted on them. Some of the components had very long procurement lead times of a year or more. To encourage its customers to communicate demands early, the PCB plant had a series of time fences that restricted the changes in order quantity and type at various lead times prior to the requested due date. However, because the company knew that long-lead-time parts would be difficult to obtain if demands were increased, customers had strong incentive to overestimate their requirements. Sure enough, when we checked the data, we found that at each time fence requirements dropped significantly (e.g., if a time fence allowed a 15 percent reduction in order quantity without cost penalty, then many orders were decreased by exactly this amount when they reached that time fence). The result was to drive excess quantities of the long-lead-time parts into the PCB plant's inventory. One remedy, as suggested above, would be to restrict customers' ability to alter orders. For instance, if the PCB plant had a frozen zone longer than the lead time of all its components, such gaming behavior would not occur. But of course it is not reasonable to impose a one year frozen zone on customers. The alternative, therefore, is to work to reduce lead times of the components so that customers will have less incentive to try to trick the system into overordering for these parts.

Finally, we observe that a sweeping policy for reducing all the factors contributing to the bullwhip effect is to eliminate whole layers of the supply chain. This is precisely what Dell Computer did with its direct marketing system in which computers were sold by the manufacturer to the customer without the use of resellers. In addition to giving Dell access to direct customer demand data, it eliminated a whole level of inventory and hence cost. This strategy played a major part in making Dell one of the most successful companies in America during the 1990s.

17.7.4 An Approximation for a Two-Level System

We now turn to a specific supply chain problem by considering a two-echelon inventory system with a single warehouse that supplies a number of facilities, which in turn supply customer demands. Assume that both warehouse and facilities make use of continuous review inventory control policies, where the warehouse uses a (Q, r) policy and the facilities use base stock policies (i.e., they replenish stock one at a time, so in effect they use (Q, r) policies with $r = 1$). This type of system makes sense for a spare parts system, where speed of delivery is crucial and volumes are relatively small. Thus, facilities are likely to receive shipments of parts from the warehouse on a frequent basis, and one-at-a-time replenishment is a practical option. This assumption may be less appropriate for retail systems, where outlets are replenished less frequently and high volumes make bulk deliveries necessary. We refer the interested reader to Nahmias and Smith (1992) for details on modeling retail systems.

The one-at-a-time facility replenishment assumption implies that demands at the facilities are passed directly back to the warehouse. This means that if demand for each part at each facility is distributed according to the Poisson distribution, then total demand at the warehouse is also Poisson-distributed. (Recall that in Chapter 2 we observed that the Poisson distribution is often a reasonable modeling assumption for representing demand processes.) This allows us to take the following approach. First we analyze the warehouse using a single-level (Q, r) model, where we fix the service level (fill rate) and compute order quantities and reorder points for each part. Then we compute the expected number of backorders outstanding at any point for each part and use this to estimate the delay that an order from a facility will experience. With this, we approximate lead times seen by the facility as the expectation of the actual delivery time from the warehouse plus this delay. Then, using these modified lead times, we apply a base stock model to each facility to compute reorder points for each part.

To develop a model, we will make use of the following notation, which is analogous to that used for the multi-item (Q, r) model above, with additional subscripts m to indicate the facility:

N = total number of distinct part types in system

M = number of facilities serviced by warehouse

$D_j = \sum_{m=1}^{M} D_{jm}$, annual demand (units per year) for part j at warehouse

ℓ_j = replenishment lead time (in days) for part j to warehouse, assumed constant

θ_j = expected demand during replenishment lead time for part j $(\theta_j = D_j \ell_j / 365)$

$p_j(x)$ = probability of exactly x demands during replenishment lead time for part j at warehouse (probability mass function)

$G_j(x) = \sum_{y=0}^{x} p_j(y)$, probability that demand for part j at warehouse during replenishment lead time is less than or equal to x (cumulative distribution function)

W_j = expected time an order for part j waits at warehouse due to backordering

D_{jm} = annual demand (units per year) for part j at facility m

ℓ_{jm} = lead time (in days) for facility m to receive part j from warehouse, assumed constant

θ_{jm} = expected demand during replenishment lead time for part j $(\theta_j = D_j \ell_j / 365)$

$p_{jm}(x)$ = probability of exactly x demands during replenishment lead time for part j at facility m (probability mass function)

$G_{jm}(x) = \sum_{y=0}^{x} p_j(y)$, probability that demand for part j at facility m during replenishment lead time is less than or equal to x (cumulative distribution function)

L_{jm} = lead time (including backordering delay) for an order of part j from facility m to be filled by warehouse, a random variable

c_j = unit cost (dollars) of part j

Q_j = order quantity for part j at warehouse (decision variable)

r_j = reorder point for part j at warehouse (decision variable)

r_{jm} = reorder point for part j at facility m (decision variable)

R_{jm} = $r_{jm} + 1$, base stock level for part j at facility m (decision variable equivalent to r_{jm})

$F_j(Q_j)$ = order frequency (replenishment orders per year) for part j at warehouse as a function of Q_j

$S_j(Q_j, r_j)$ = fill rate (fraction of orders filled from stock) of part j at warehouse as a function of Q_j and r_j

$B_j(Q_j, r_j)$ = average number of outstanding backorders for part j at warehouse as a function of Q_j and r_j

$I_j(Q_j, r_j)$ = average on-hand inventory level (in units) of part j at warehouse as a function of Q_j and r_j

Warehouse Level. We can solve the warehouse problem (i.e., compute Q_j and r_j for all parts) by using any of the approaches given earlier for the single-level problem. That is, we could use a cost model in which we specify a fixed order cost A and either a backorder cost b or a stockout cost k. Or we could use a constrained model in which we specify constraints on the average number of orders per year F and either the fill rate S or the average backorder level B. Typically, it makes more sense to use a model based on a backorder cost or constraint, rather than one based on fill rate, since the reason for holding inventory in the warehouse is to minimize delay seen by the facilities (and hence the customers).

Regardless of what model we use, we will wind up with a set of Q_j and r_j values, which can then be used to compute F_j, S_j, B_j, and I_j for all parts $j = 1, \ldots, N$ using the functions developed in Chapter 2. We will use these as inputs to the calculations at the facility level.

Facility Level. Observe that the expected time (in days) an order from a facility waits at the warehouse due to backordering is

$$W_j = \frac{365 B_j(Q_j, r_j)}{D_j} \tag{17.14}$$

Notice that this is nothing more than an application of Little's law to the backorders (i.e., the wait is analogous to cycle time, the backorder level is analogous to WIP, and the demand rate is analogous to throughput). Hence we can estimate the mean effective lead time (in days) for part j to facility m as

$$E[L_{jm}] = \ell_{jm} + W_j \tag{17.15}$$

We could just act as though this mean lead time were a constant and use it in the base stock model to compute performance measures for the facilities. Indeed, researchers have shown that treating these lead times as if they were equal to their means (that is, L_j) can yield reasonable results (see Sherbrooke 1992). However, it is clear that L_{jm} is a random variable that could exhibit a great deal of variability. When an order from the facility to the warehouse finds stock available, $L_{jm} = \ell_{jm}$. But when an order finds the warehouse in a state of stockout, then L_{jm} could be much longer than this. Computing the exact distribution of the effective lead time seen by a facility is complicated (see de Kok 1993). But we can incorporate the effect of lead time variability in an approximate way.

Technical Note

To approximate the variance of the effective lead time of an order from a facility to the warehouse, suppose that there are only two possibilities: Either the order sees no delay and the lead time is ℓ_{jm}, or it does encounter a stockout delay and has lead time $\ell_{jm} + y$, where y is a deterministic delay Since the probability of stockout is $1 - S_j$ (we will omit the dependence of S_j and B_j on Q_j and r_j for notational convenience), we know that

$$E[L_{jm}] = S_j\ell_{jm} + (1 - S_j)(\ell_{jm} + y) = \ell_{jm} + (1 - S_j)(y) \tag{17.16}$$

But in order for this to match Equation (17.15), we must have

$$y = \frac{W_j}{1 - S_j} \tag{17.17}$$

To calculate the variance of L_{jm}, we first compute

$$E[L_{jm}^2] = S_j\ell_{jm}^2 + (1 - S_j)(\ell_{jm} + y)^2 \tag{17.18}$$

and then

$$\begin{aligned}\mathrm{Var}(L_{jm}) &= E[L_{jm}^2] - E[L_{jm}]^2 \\ &= S_j(1 - S_j)y^2 \\ &= \frac{S_j}{1 - S_j}W_j^2\end{aligned} \tag{17.19}$$

The standard deviation of the effective lead time to the facility (in days) is therefore approximately equal to

$$\sigma(L_{jm}) = \sqrt{\frac{S_j}{1 - S_j}}W_j \tag{17.20}$$

We can use $E[L_{jm}]$ and $\sigma(L_{jm})$ in a base stock model for each part j at facility m to compute a base stock level R_{jm}.

Integrating Levels. There are two issues to be addressed to coordinate the two levels: the model to use at the warehouse level and the parameters to use in the model. Once we have chosen these, the above method for modeling the facility level will adjust the base stock levels for facilities accordingly.

In a multiechelon spare parts supply chain, the most natural model for the warehouse level is the backorder model. The reason is that service to the customer is closely related to delay caused by part outages. Hence, the key measure of service at the warehouse is time delay, which we have seen is proportional to backorder level. Therefore, a logical choice of a warehouse model is the backorder (Q, r) model with a constraint on backorder level. We can use the previously described algorithm to compute the order quantities Q_j and reorder points r_j for the warehouse. Equivalently, we could use the backorder model with a backorder cost b instead of a constraint on backorder level. However, it is usually more intuitive to set a target backorder level (or time delay) constraint than it is to specify a backorder cost.

In other multiechelon supply chains, such as retail systems, customer service may be more appropriately measured by the fill rate. For instance, if orders that cannot be filled immediately at the warehouse are either lost or shunted to a (more expensive) third party, then fill rate makes perfect sense as the service measure at the warehouse. However, we would need to modify the model to account for lost sales or a different dependence of the lead times on the warehouse service level.

Once we have a model for the warehouse level, we need to specify its parameters. If we use the constrained backorder model, then the key decisions concern what to use for the order frequency target F and the target backorder level B. The order frequency target can be selected directly by considering the capacity of the warehouse procurement system and hence the number of replenishment orders that it can accommodate annually. Alternatively, we could specify a fixed cost of placing an order A and use this in the multipart EOQ formula (17.7) to compute order quantities.

Selecting the target backorder level is more difficult. How many backorders are allowable at the warehouse depends on what this does to performance at the facilities. Therefore, it is almost impossible to specify a backorder level target a priori. Instead, what we should do is to think of this backorder level target as a variable that we can adjust to seek the best overall system performance. Specifically, we solve the warehouse level using a given backorder level target. Then we solve the facility level so as to achieve the desired backorder level or fill rates at the facilities and observe the inventory holding cost (or investment). Finally we go back and try a different backorder level target at the warehouse and resolve both levels to see if the same performance at the facilities can be achieved with a lower inventory cost. Changing the backorder level target will alter the balance of inventory at the warehouse versus the facilities. The search for a backorder target that achieves the optimal balance can be automated within a spreadsheet or other optimization routine.

Example:
We conclude this section with a two-echelon example. Because our purpose is to highlight the relationship between levels, we will keep things simple by looking at only a single part.

Suppose the example we solved for Jack, the maintenance department manager (Chapter 2, Table 2.6), actually represents the warehouse in a two-echelon supply chain. Jack stocks spare parts at the warehouse in order to supply various regional facilities, which provide the parts for use in actual machine repair. Omitting the subscripts j because this is a single-part example, we see the key data for the warehouse are $D = 14$ parts per year, $Q = 4$, and $r = 3$. Recall that we computed the order quantity $Q = 4$ and reorder point $r = 3$ in Chapter 2 by using the backorder cost model (assuming a fixed setup cost of $A = \$15$ and a backorder cost of $b = \$100$). But we could have just as easily have used a constrained model with constraints on order frequency F and backorder level B.

Now let's extend this example by looking at a single facility with $D_m = 7$ (i.e., the facility accounts for one-half of the annual demand seen by the warehouse). From the calculations in Chapter 2, we know that $B(4, 3) = 0.0142$ unit, so the average time a replenishment order waits due to lack of inventory is

$$W = \frac{365B(4,3)}{D} = \frac{365(0.0142)}{14} = 0.3702 \text{ day}$$

Supposing that the actual delivery time to receive a part from the warehouse is one day, the expected lead time for a part is

$$E[L_m] = 1 + 0.3702 = 1.3702 \text{ days}$$

and hence expected demand during replenishment lead time to the facility is

$$\theta_m = \frac{1.3702 \times 7}{365} = 0.0263 \text{ unit}$$

Also from our previous calculations in Chapter 2, we know that the fill rate is $S(4, 3) = 0.965$. Hence, the standard deviation of replenishment lead time is

$$\sigma(L_m) = \sqrt{\frac{S}{1-S}}W = \sqrt{\frac{0.965}{1-0.965}}(1.3702) = 1.944 \text{ days}$$

Assuming that demand at the facility level is Poisson, we can use Equation (2.58) to compute the standard deviation of lead time demand as

$$\sigma_m = \sqrt{\theta_m + \left(\frac{D_m}{365}\right)^2 \sigma(L_m)^2} = \sqrt{0.0263 + \left(\frac{7}{365}\right)^2 (1.944)^2} = 0.166 \text{ unit}$$

Note that in this example $\sigma_m = 0.166$ is very close to $\sqrt{\theta_m} = \sqrt{0.0263} = 0.162$. The reason is that the inflation factor in Equation (2.58) is relatively small. This implies that lead time demand is very close to Poisson. Hence, we can use the Poisson formulas to approximate the service that results from various base stock levels.[10] For instance, if we set the reorder point for the facility equal to $r_m = 0$, then the fill rate is given by

$$\begin{aligned} G_m(r_m) &= \sum_{y=0}^{r_m} p(y) = p(0) \\ &= \frac{\theta_m^0 e^{-\theta_m}}{0!} = e^{-0.0263} \\ &= 0.974 \end{aligned}$$

If we increase the reorder point to $r_m = 1$, then service increases to 0.997. So, depending on the criticality of this part at the facility, it looks as if a reorder point of zero or one will be appropriate.

17.8 Conclusions

Inventory management is as old as manufacturing itself. Analytical approaches to inventory control date back to the scientific management era (i.e., the early 20th century) and are among the earliest examples of operations research/management science. Despite this, the field continues to evolve. Even techniques as old as the EOQ and (Q, r) models are experiencing breakthroughs (e.g., new algorithms and use in multiechelon supply chains). Thus, it appears that the final word on inventory and supply chain management is far from written. The models presented in this chapter provide reasonable approaches to some settings, but better methods and extensions to new settings will undoubtedly evolve. This means that inventory will be an area ripe for continual improvement and that manufacturing managers will need to continue learning new tricks in this important field.

In the meantime, the following tips are worth keeping in mind:

1. *Understand why inventory is being held.* Different types of inventory are held for different reasons, some conscious and others unconscious. Rigorously asking the question of why each type of inventory is held in a given system can reveal inefficiencies that are being taken for granted.

2. *Look for structural changes.* Fine-tuning a supply chain through the use of sophisticated models is fine. However, really big improvements are likely to require structural changes. For instance, changing from a strategy of stocking FGI to one of stocking semifinished product and producing to order might have a dramatic effect on total inventory investment. Similarly, eliminating the central warehouse and stocking all spare parts at regional facilities could produce a substantial improvement in customer service with no increase in inventory. The specific changes that are possible depend on

[10] Since the actual variability is slightly greater than the Poisson distribution, actual service will be slightly lower than predicted by the Poisson formulas.

the system. The key to identifying them is to take for granted as little of the status quo as possible.

3. *Use emprical evaluation procedures.* Any model is based on simplifying assumptions (e.g., steady state, Poisson demand), and input data are approximate at best. Thus, the best analysis can do is to help us find a reasonable policy (finding the "optimum" is out of the question) and examine tradeoffs. Given this, we should be careful to supplement analysis with empirical observation and feedback. Examples of parameters we should monitor include (1) service levels, to compare with those predicted by our models and to determine whether policy changes are needed; (2) minimum inventory levels and stockout frequency of stock in raw materials and FGI, to determine whether we are carrying insufficient or excessive safety stock; and (3) queue lengths and starvation time at key workstations, to detect excessive or insufficient WIP. Many other measures may make sense depending on the system. The important thing is to identify a few key measures and set up an adequate data collection and interpretation system for them.

4. *Cycle time reduction is crucial.* Little's law tells us that where there is WIP, there is cycle time. So WIP reduction and cycle time reduction are virtually synonymous. But even more importantly, reduced cycle times make it possible to rely less on distant forecasts in the purchase of components and the scheduling of work. The net result, therefore, is smaller raw materials and FGI levels, as well as less WIP.

5. *Coordinate levels in multiechelon supply chains.* Inventory management grows more complex when stock is held at multiple levels. In addition to managing each level efficiently, it is critical to make sure that performance at the separate levels supports overall system efficiency. The bullwhip effect is an important example of how myopic control of the separate levels can cause huge problems for the system as a whole. To avoid these, it is important to analyze the supply chain as a whole, rather than as separate parts, share common data (e.g., retail demand data) wherever possible, and streamline the supply chain to avoid unnecessary complexity.

6. *Coordinate incentive systems with objectives.* It is well and good to set up an inventory management system with specific performance goals in mind. However, any such system will rely on people to make it work. Therefore, if the reward structure does not support the system goals, it is unlikely to work. (Recall the personnel law: *people, not organizations, are self-optimizing.*) For example, we recently worked for a company with a multiechelon supply chain in which facilities were evaluated primarily in terms of customer service but, in the name of inventory efficiency, also had their inventory levels audited once per month. Predictably, facility managers had a tendency to hoard inventory (i.e., carry more than the recommended levels) all month. Right before the end-of-month audit, they would send the excess back to the distribution center. Once the audit was completed, they would order back up to their "excessive" levels. The effect was to destroy any balance between inventory and service. Clearly, no modeling or analysis effort could correct this problem. Only revising the facility evaluation procedure (e.g., by using ratings that combine service with inventory level, where inventory is measured continuously or randomly in units of dollars) could rationalize the facility inventory levels.

Discussion Point

Suppose a manufacturer of electric mixers sells virtually identical models to several retailers. The major differences between models are the boxes (which are printed with glossy pictures of the mixer and the house brand of the retail outlet) and the paper inserts (which include instructions and retailer-specific information). Demand is strongly seasonal (i.e., peaking around Christmas), so the firm follows a strategy of building inventory (FGI) in the off season. The problem is that while forecasts for total volumes are typically reasonable, the forecasts for individual retailers can be awful. The result is that the firm is frequently short of fast-selling models and awash in slow-moving ones. What general strategies might the firm consider to improve customer service and reduce FGI?

Study Questions

1. Why might the EOQ model be better-suited to purchased parts than to internally manufactured products?
2. How can cycle time reduction reduce raw materials, WIP, and FGI?
3. In general, WIP reduction techniques are also lead time reduction techniques, but the reverse is not always true. List some lead time reduction techniques that do not reduce WIP.
4. What causes large inventories of unmatched parts at an assembly operation? What measures might we consider to address such a situation?
5. What is the difference between type I and type II service? What is the rationale for using type I service in a (Q, r)-type model?
6. Why do we use approximations for fill rate and backorder level in the algorithms for computing Q and r, but check the constraints on these measures against the exact formulas?
7. Suggest appropriate performance measures for evaluating the efficiency of raw materials, WIP, FGI, and spare parts in a manufacturing system.
8. List some examples of arborescent multiechelon supply chains. Can you think of a system that has the reverse of the anborescent structure (i.e., so that many high-level sites supply a few middle-level sites, which in turn supply a single low-level site)?
9. What are the four main causes of the bullwhip effect in multiechelon supply chains? Which causes are likely to have the largest effect in each of the following systems?
   a. A consumer products distribution network, consisting of the manufacturing plant, regional warehouses, and retail outlets.
   b. A spare parts network, consisting of a main distribution center, regional facilities, and customer sites.
   c. A military supply network, consisting of a central warehouse, regional depots, and field usage sites.
10. List some supply chains in which holding the bulk of the stock at the demand level (e.g., at retail outlets) and making use of lateral transshipments might make sense.
11. What incentive or reward system changes might be required to effectively reconfigure a multiechelon supply chain to do away with the central warehouse and store all inventory at regional facilities?

Problems

1. CMW, a custom metalwork shop, makes a variety of products from three basic inputs—bar stock, sheet metal, and rivets—which are purchased in bars, sheets, and kits (boxes of 100), respectively. Projected use and cost of these raw materials for the upcoming year are as follows:

Part	Use (1,000 units/yr)	Cost ($/unit)
Bar stock	120	40
Sheet metal	400	20
Rivet kits	1000	0.5

The shop estimates that issuing a purchase order for any type of material costs $100 and uses an interest rate of 15 percent to calculate holding costs.

a. Assuming steady use throughout the year, estimate the purchasing plus holding cost if all products are purchased four times per year. What happens to cost if we purchase each product 12 times per year?

b. What are the "optimal" order frequencies if we use the EOQ model separately for each product? How many total purchase orders must be placed under this policy?

c. Use the EOQ model to compute order quantities for each part and adjust the fixed cost of placing an order until the *average* order frequency is 12 times per year. How does the holding cost compare to that in part *a* where all parts are ordered 12 times per year?

2. Rivethead Charlie is in charge of the raw materials crib at a facility that manufactures specialized camping gear. In one part of the crib, Charlie stocks connectors. These are not included on the bills of material for the end items, but instead are ordered according to Charlie's "two-bin" system. Under this system, Charlie maintains two bins for each type of connector that hold 1,000 units each. Whenever one bin of a connector becomes empty, Charlie opens up the second bin and orders a refill (that is, 1,000 units) to replenish the first bin. The two most common connectors are rivets, which are used at an average rate of 2,000 per month, and screws, which are used at an average rate of 500 per month. The replenishment lead time from the supplier is two weeks (one-half month), and the unit cost is $0.10 for both rivets and screws. You can assume that demand (use in the manufacturing process) is Poisson for both types of connector.

   a. Note that Charlie is following a (Q, r) policy. What are Q and r for rivets and screws under his policy?

   b. What are the average fill rate and inventory investment (total for both parts) under Charlie's policy?

   c. A summer intern suggests that Charlie should use "days of supply" to set the sizes of the bins, rather than a fixed size of 1,000. What would be the (Q, r) policy that would result if Charlie used bins sized to hold a one month supply of parts? What are the average fill rate and inventory investment under this new policy?

   d. Suppose Charlie uses a two-bin policy in which bins hold five weeks (1.25 months) of supply. What are Q and r for rivets and screws, and what are the average fill rate and inventory investment? What do the results of parts *c* and *d* say about the efficacy of using the days-of-supply approach to bin sizing? Is the intern's suggestion a good one?

   e. What type of policy might be better than a two-bin policy, with or without the days-of-supply modification?

3. Stock-a-Lot maintains inventories of parts to support repairs of manufacturing equipment. For a subset of its parts, the expected use, unit cost, and replenishment lead time for the upcoming year are forecast as follows:

Part	Use (units/yr)	Cost ($/unit)	Lead Time (months)
1	5	1,000	1
2	10	100	2
3	5	200	6
4	20	1,000	1
5	50	50	3

a. Find order quantities that make the average order frequency equal to five times per year, by adjusting the fixed order cost and using the EOQ model.

b. Using the order quantities from part *a*, compute the reorder points so that the fill rate is 95 percent for all parts; and compute the average inventory investment.

c. Using the order quantities from part *a*, compute the reorder points that achieve an *average* fill rate of 95 percent, by adjusting the stockout cost in the stockout model algorithm.

d. Compute the average backorder level resulting from the solution to part *c*. Using the backorder model algorithm and the order quantities from part *a*, find the reorder points that attain the same backorder level as part *c*. How does the total inventory investment compare to that from part *c*?

4. Reconsider the Stock-a-Lot problem, and suppose now that the warehouse supplies several regional facilities. Assume the warehouse is stocked according to the policy computed in part *c* of Problem 3. Consider a single facility supplied by the warehouse that has 12-hour actual delivery times and a demand rate for part 4 of 10 units per year. Compute the following for part 4.

   a. Find the expected number of outstanding backorders at the warehouse.

   b. Determine the expected effective lead time to the facility.

   c. Treating demand at the facility as Poisson, find the minimum base stock level for part 4 at the facility that achieves a target service level of 99 percent.

5. A&T, Inc., has a spare parts system that corresponds to the example depicted in Figure 17.4.

   a. A&T's current stocking policy has resulted in an average order frequency of $F = 12$, a fill rate of $S = 0.85$, and an inventory investment of $2,500. Comment on the quality of the policy. If you were to encounter a situation like this in practice, what system elements would you look at in the hope of making improvements?

   b. The president of A&T has demanded a system with a fill rate of $S = 0.95$ and inventory investment of no more than $1,000. What can you say about the feasibility of this demand? How could you respond to it?

6. Windsong, a novelty store that sells wind chimes and related items, stocks the popular "Old Ben" model. Sales are steady at a rate of one per day (365 per year), and demand can be regarded as Poisson. Windsong purchases Old Bens, along with other products from a supplier that makes daily deliveries. Hence, Windsong uses a base stock policy for its products.

   Suppose that the supplier has set its stocking policy such that the fill rate and average backorder level for Old Bens are 89.7 percent and 0.465 day, respectively. Replenishment lead time is seven days.

   a. What is the expected demand during replenishment lead time when delays by the supplier are taken into consideration?

   b. What is the standard deviation of lead time demand? Is it more or less variable than Poisson?

   c. If we assume demand is Poisson, what fill rate will result from a base stock policy with a reorder point of 10? Will the actual fill rate be higher or lower than this?

CHAPTER

18 Capacity Management

You can't always get what you want.
No, you can't always get what you want.
But if you try sometimes, you just might find
You get what you need.
Rolling Stones

18.1 The Capacity-Setting Problem

Choices about how much and what type of capacity to install have a strong direct influence on a firm's bottom line. Additionally, because **capacity planning** is at the top of the plant planning hierarchy (see Figure 13.2), capacity decisions have a major impact on all other production planning issues (e.g., aggregate planning, demand management, sequencing and scheduling, shop floor control). In this chapter we invoke factory physics concepts to translate strategic capacity decisions into specific tactical terms. Our goal is to provide a framework for capacity planning that explicitly recognizes its impact on the overall plant management process.

18.1.1 Short-Term and Long-Term Capacity Setting

There are many times in the life cycle of a manufacturing facility when it makes sense to adjust capacity. Most often, the motivation is to accommodate a change in the total volume or the product mix of demand. In the short term, the facility can address demand changes through the use of overtime, addition or deletion of shifts, subcontracting, and workforce size changes. These policies were discussed in Chapter 16 in the context of aggregate planning; they are clearly options in capacity planning as well.

Some of these short-term options may also be viable as long-term policies. For instance, we could run three shifts or subcontract part of or all production on a semipermanent basis. Of course, if we outsource manufacturing of a product to a vendor on a long-term basis, the vendor might eventually decide to sell it directly and become a competitor. Fortunately, however, there are barriers to entry that often prevent this. For example, nonmanufacturing factors such as rights to a recognizable brand name or possession of an effective delivery/service network can be critical. Even if eventual competition is not a serious risk, relying on vendors to manufacture parts or products makes them a significant partner in the quality management process, as we discussed in Chapter 12. Without measures to ensure vendor quality, the decision to outsource manufacturing can seriously hamper a firm's ability to control its destiny.

In the long term, we must go beyond these short-term options and consider permanent equipment, or "bricks and mortar" changes. These involve either major changes to an existing facility or construction of a new facility altogether. In some cases, a firm can permanently increase capacity by redesigning a product, using design for manufacture (DFM) approaches (see Turino 1992, Chapter 7 for a discussion). More frequently, however, the change must come from either adding machines or processing stations or making permanent changes in the productivity of existing equipment or procedures.

18.1.2 Strategic Capacity Planning

Before a firm can consider how much and what type of capacity to install, it must articulate a capacity strategy. Such a strategy hinges on decisions that are very close to the firm's core business plan. For instance, it may need to decide whether to enter a new market, whether to remain in an existing market, to lead or follow in the product innovation process, to make or outsource a product, what segment of the market to pursue, and many other questions. Taken together, these questions are tantamount to the fundamental strategic question of "What business are we in?" which lies beyond the scope of factory physics. The laws of physics can tell us how a particular physical system will behave but not what system we should be interested in. Similarly, the laws of manufacturing can help us design systems to attain specific objectives but cannot tell us what our objectives should be. Therefore, for the purposes of our discussion, we will assume that the above strategic decisions have been made and that the issue is how to evolve a capacity plan to support them.

Once we have decided that we need to add capacity, there are several issues to address:

1. *How much and when should capacity be added?* Should additions be made only when demand has already developed (when we are already losing sales), or in anticipation of future demand? If we don't anticipate demand, should we fill in the overcapacity periods by using short-term measures such as overtime or subcontracting? If we decide to anticipate demand, how far into the future should we try to cover? Adding large increments will satisfy demand farther into the future, will cause fewer construction disruptions, and can take advantage of economies of scale. However, large increments also imply poorer equipment utilization and greater exposure to risk. (What if the forecasted demand does not materialize?) The appropriate approach also depends on the production technology involved. For example, steel mills must generally add capacity in large units in the form of new furnaces or rolling mills, while a metalworking job shop can add small increments of capacity by adding individual machines. See Freidenfelds (1981) for an analysis of these issues.

2. *What type of capacity should be added?* The size of the capacity increment we can add also depends on the flexibility of the equipment we choose. If machines purchased now can be adapted to new products that will be introduced in the future, the risk of installing more capacity than currently needed is substantially less. In today's environment of rapid product change, product lifetimes are often less than the lifetimes of the production equipment; consequently, this type of flexibility has become a key

consideration in choosing new capacity. See Sethi and Sethi (1990) for a review of the different types of flexibility in manufacturing systems.

3. *Where should additional capacity be added?* Should we add capacity by expanding an existing facility, or should we build a new one? Although it is often more expensive to build a new facility than to expand an existing one, the new facility often affords new marketing and distribution efficiencies, for instance by being closer to either suppliers or customers. See Daskin (1995) for models of the facility location problem.

An important strategic concept is known as **production economies of scale.** The basic idea is that unit costs are typically (but not always) less for a large plant than for a small one. Hayes and Wheelwright (1984) discuss three different economies of scale: short-, intermediate-, and long-term.

Short-term economies of scale arise from the fact that in the very near term, many manufacturing costs are fixed. Although adjustable in the longer term, the production facility, its labor force, management, insurance cost, property taxes, and so on, for any given day, are all *fixed.* The cost of these does not depend on production volumes. Indeed, in the near term, the only true *variable* costs are material, some utilities, and some wear on machines. We can express cost per unit as

$$\begin{aligned}\text{Unit cost} &= \frac{\text{Fixed cost} + \text{Variable cost}}{\text{Throughput}} \\ &= \frac{\text{Fixed cost}}{\text{Throughput}} + \text{Variable unit cost}\end{aligned}$$

Thus, in the short term, unit cost decreases as throughput increases.

Intermediate-term economies of scale depend on the run lengths used in production—the number of units of a product that are produced before the facility switches to another product. Given the changeover cost and run length of a particular product, unit cost can be expressed as

$$\text{Unit cost} = \frac{\text{Changeover cost}}{\text{Units per run}} + \text{Running cost per unit}$$

In this case, labor might or might not be fixed. Run lengths can be affected by setting up less frequently (facilitated through setup reduction), by dedicating equipment (so that some product families can be continually run without changing over), and by using specialized equipment (e.g., flexible manufacturing systems). Of course, some of these options can result in larger inventories, as we discussed in Part II.

Long-term economies of scale are functions of plant equipment itself. Economists have long noted that the cost of equipment tends to be proportional to its surface area, while capacity is more closely proportional to volume. To illustrate the implications of this, suppose the equipment is a cube with side length ℓ. Then we can express cost as

$$K = a_1\ell^2$$

and capacity as

$$C = a_2\ell^3$$

where a_1 and a_2 are proportionality constants. To express cost as a function of capacity, we solve for ℓ in terms of C, and we get $\ell = a_3C^{1/3}$, with a_3 representing another constant; then we substitute into the cost expression. This yields

$$K(C) = aC^{2/3}$$

where, again, a is a proportionality constant.

For general (non-cube-shaped) equipment, cost as a function of the capacity can be approximated by

$$K(C) = aC^b$$

where b is typically between 0.6 and 1.

We can now express cost per unit as

$$\text{Unit cost} = \frac{K(C)}{C} = aC^{b-1}$$

Since b is usually less than one, this implies that unit cost tends to decrease with capacity. That is, large plants are more efficient than small ones.

In practice, economies of scale frequently do enable bigger plants to achieve lower unit costs, but not always. There can also be **diseconomies of scale** that cause the organization to lose efficiency as it becomes larger. One place this happens is in distribution. A small compact cell has less material handling than a large plant composed of many process centers. While process centers in the large plant may be more efficient than the single stations of which the cell is composed, jobs must also be moved greater distances. This increases material handling and cycle times. Also since large manufacturing plants typically serve larger areas than small ones, their freight costs are typically higher. In the case of bulky commodity products like bricks, the most profitable plant size may be quite small.

Another form of diseconomy of scale is due to bureaucratization. As the size of the operation increases, so does the required amount of supervision and support. To keep the span of control manageable, the large firm adds layers of management, which further decreases communication effectiveness. This can lead to compartmentalization and turf wars. If not managed carefully, such diseconomies can be very destructive.

Finally, larger plants naturally create more risk. Natural disasters such as earthquakes, fires, floods, and hurricanes will obviously have a greater negative impact on the company if they strike a single large plant than if they affect a single small facility among many. Similarly, poor management, strikes, and the like are more disruptive if the company capacity is concentrated than if it is distributed.

A natural question arises in this context: What is the optimal plant size? This question is largely one of strategy, which is beyond the scope of this book. Moreover, since it involves many firm-specific issues, a general-purpose answer is not possible. The above discussion gives a preliminary overview of the issues to be considered. More detailed treatments are available in the manufacturing strategy literature (e.g., see Hayes and Wheelwright 1984; Schmenner 1993).

In keeping with our focus on plant management, we will assume that the size of the facility has already been determined on the basis of strategic considerations. Thus, we will consider the problem of how to change capacity within a plant to attain a specified set of objectives. In particular, we examine two scenarios: building a new facility and changing an existing one.

18.1.3 Traditional and Modern Views of Capacity Management

To frame the capacity-planning problem at the plant level, it is useful to distinguish between the **traditional** and the **modern** views of the role of capacity (Suri and Treville 1993). The traditional view is based on the interpretation of manufacturing efficiency shown in the left portion of Figure 18.1. Here, the only question is whether there is enough capacity to meet a particular throughput target, and the answer is either yes or no. If utilization is below capacity, then production is feasible; otherwise, it is infeasible.

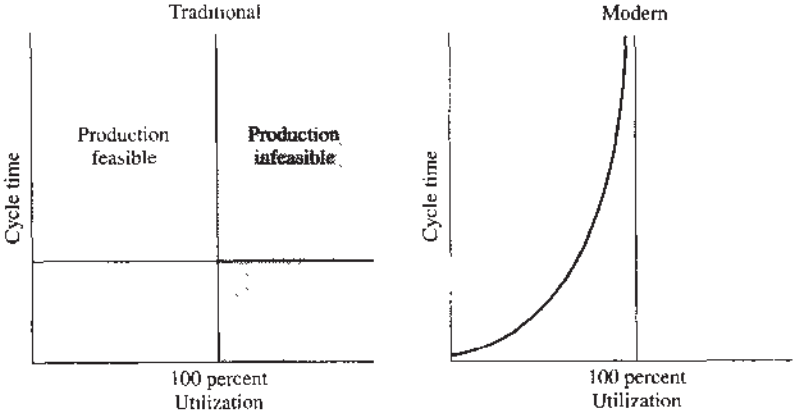


Figure 18.1

Traditional versus modern views of capacity planning

The modern view, which is more realistic and consistent with the principles of factory physics, holds that lead times and WIP levels grow continuously with increasing utilization; this is shown in the right side of Figure 18.1. In this view, there is no one point where production is infeasible. Instead a continuum of decreasing responsiveness occurs as capacity is utilized more heavily.

These two views imply very different approaches to the design of production lines. The traditional view suggests selecting a set of machines that have sufficient capacity, at the lowest possible cost. But doing this usually leads to problems when the line goes into production. We have encountered many plants with lines consisting of machines, each of which has rated capacity above the desired rate, but which consistently fall well short of their throughput targets. (The reader who has absorbed the factory physics principles of Part II should have a pretty clear idea of why such lines fail to meet throughput goals.)

The modern view affords a much richer interpretation of the capacity issue. Since capacity is more than a simple yes-or-no question, we must consider other measures of performance in addition to cost and throughput. WIP, mean cycle time, cycle time variance, and quality are all affected by capacity decisions. If we can state our objectives in terms of these measures, then we can formulate the capacity-planning problem very simply (solving it, however, is a different matter) as follows:

For a fixed budget, design the "best" facility possible.

This formulation is imprecise since what is "best" is difficult to define because we usually have more than one objective. For instance, is a line with low throughput and low cycle time better or worse than one with higher throughput and higher cycle time? As we discussed in Chapter 6, we get around the problem of dealing with multiple objectives by using the technique of **satisficing,** that is, by selecting one measure as the objective and fixing the remainder as constraints. In this way, the problem is divided into a **strategic** problem that defines one or more **tactical** problems. The strategic problem might be to choose how much capacity to have, how long cycle times should be, what types of capacity to use, what throughput is required, and so on. The tactical problem is then to minimize cost or some other quantity subject to the constraints imposed by the strategic problem. This approach of higher-level problems providing constraints for lower-level ones was discussed in Chapter 6.

One formulation would be to maximize throughput subject to a budget constraint and, possibly, constraints on WIP and cycle time. Another would be to minimize cycle time subject to constraints on budget and throughput. Still another would be to minimize cost subject to constraints on throughput, cycle time, and WIP. Which is best depends on the circumstances. If, on one hand, we are concerned with improving an existing line and have a fixed budget to spend, then the formulation to optimize something (maximize throughput or minimize cycle time) subject to a budget constraint makes perfect sense. If, on the other hand, we are designing a new line to achieve given performance specifications, then minimizing cost subject to constraints on things like throughput and cycle time is appropriate.

Regardless of the formulation chosen, we can use the resulting model to examine important tradeoffs. For instance, if we use a model to minimize cost subject to constraints on throughput and cycle time, we can vary the levels of the throughput and cycle time constraints to see how cost changes. The result will be curves of throughput versus cost and cycle time versus cost, both of which are useful in deciding whether our initial strategic specifications were reasonable.

In addition to focusing on the *optimality* of capacity decisions, we must be sensitive to their *robustness*. The requirements we specify today may be quite different from our requirements in the future. It is sometimes a good idea to spend a bit more money up front (e.g., on a capacity cushion, or on more expensive but more flexible equipment) to cover future contingencies. We can consider such options by examining various demand scenarios in the model. However, we must take care not to overbuild for the sake of robustness. One of the reasons that wafer fabrication facilities are enormously expensive is that they are designed in the hope of making almost anything that might be desired in the near future. Because technological uncertainty in semiconductor manufacturing is extremely high, this requires installing the very latest leading-edge (or "bleeding-edge") equipment.

For the remainder of this chapter, we will focus on the problem of minimizing the cost of installing or changing a line, subject to various performance constraints. We have chosen this particular formulation for the following reasons: (1) It is the most natural framework for considering the new line design problem, and (2) it is well adapted to generating cost-versus-performance tradeoff curves. However, one can easily analyze other formulations (e.g., to minimize cycle time subject to throughput and cost constraints) using the tools and techniques we present here.

18.2 Modeling and Analysis

We have relied heavily on models throughout this book, primarily because models force us to think carefully about the systems we are studying and help us develop intuition about how they behave. But at the practical level, without some form of model, either explicit or implicit, one cannot do analysis at all. Accounting, marketing, finance, quality control, and virtually all other business functions rely on models to interpret data, predict performance, and evaluate actions. Happily, the models upon which we rely to address the capacity-planning problem are largely the same as those we used in Part II to explain the concepts of factory physics. In particular, we use the queueing network representation of a manufacturing line to develop capacity analysis tools. Although we adhere to the basic formulas introduced in Part II, there is a large literature on these tools, and we refer the interested reader to Buzacott and Shanthikumar (1993), Suri et al. (1993), and Whitt (1983, 1993) for more details.

For clarity, we concentrate our analysis on a single line and regard the remainder of the production facility as fixed. We assume that the line has M workstations and that the "manufacturing recipe" is given—that is, the operations required at each station to produce the part or product are set in advance. We consider here only the case in which the line produces a single product, although we can accommodate the multiple-product case by attributing the variability due to different processing times of different products at the stations to the natural variability at the process centers (i.e., by inflating the coefficient of variation of the effective processing times). We number the stations $1, 2, \ldots, M$, where jobs arrive to station 1, which feeds them to station 2, which feeds them to station 3, and so on. In this discussion we do not consider rework or branching routings, although these can be accommodated by using more sophisticated versions of the queueing network models (see Suri et al. 1993).

For each station there are a number of different **technology options**, consisting of specific configurations of machines and/or operating policies, from which to select. These options might include different models of machines from various equipment vendors. They might also include a machine with and without a kit of field replacement parts, where the option with the replacement parts has shorter repair times but higher cost than the option without them. Notice that this definition makes identifying an appropriate set of technology options more than a matter of collecting data from equipment vendors. We must make use of our factory physics intuition from Part II to recognize options like field replacement parts that are potentially attractive. We assume here that a reasonable set of technology options can be generated and that cost, capacity, and variability parameters can be estimated for each option.

To keep the number of technology options and the analysis manageable, we assume that no mixing of machine types is allowed at multimachine stations. In other words, if the line requires three lathes and we have chosen the South Bend X-14 as our model, we will use three South Bend X-14s. We cannot use two South Bend X-14s and one Peoria P1000. This restriction is likely to be satisfied naturally in new lines, since we are unlikely to want to deal with two equipment vendors when we can deal with only one. In retrofit situations, it may not be literally satisfied, but is frequently not a major problem from a modeling perspective.

Each option at each station is described by five parameters:

t_e = mean effective process time for machine, including outages, setups, rework, and other routine disruptions

c_e = effective coefficient of variation (CV) for the machine, also considering outages, setups, rework, and other routine disruptions

m = number of (identical) machines at station

k = cost per machine

A = fixed cost of machine option

The total cost of installing the option is given by $A + km$. Thus, if it costs \$75,000 to install one machine and \$125,000 to install two machines, then A = \$25,000 and k = \$50,000. The idea here is to allow us to represent the costs of activities that need only be done once, regardless of the number of machines installed, such as modifying the electrical or ventilation systems or reinforcing the floor.

We described how to compute t_e and c_e^2 from more basic parameters in Chapter 8. Here we assume that these have already been computed for each option. However, it may be useful to examine the more basic parameters (MTTR, MTTF, etc.) to suggest other technology options.

To formulate constraints for the model, we assume that strategic decisions have been made regarding the overall performance of the line, which establish the following:

$$\text{TH} = \text{required throughput}$$

$$\text{CT} = \text{maximum total cycle time}$$

Then, using the above parameters and a description of the arrival process to the line, we compute the following for each station in the line:

$$\begin{aligned} u(m) &= \text{utilization of station with } m \text{ machines installed} \\ \mathrm{CT}(m) &= \text{cycle time at station with } m \text{ machines installed} \\ c_a &= \text{CV of arrivals to station} \\ c_d &= \text{CV of departures from station} \end{aligned}$$

The formulas for computing u and CT are familiar from Part II and can be expressed as

$$u(m) = \frac{r_a t_e}{m} \tag{18.1}$$

$$\mathrm{CT}(m) = \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u^{\sqrt{2(m+1)}-1}}{m(1-u)}\right) t_e + t_e \tag{18.2}$$

The squared coefficient of variation (SCV) of the arrivals c_a^2 is specified as a parameter for station 1, and for subsequent stations is equal to the SCV of the departures from the previous station. That is, letting $c_a^2(i)$ and $c_e^2(i)$ represent the SCV of the arrival times and effective processing times at station $i (i = 1, \ldots, M)$, respectively, we have

$$c_a^2(i) = \begin{cases} \text{a specified constant} & i = 1 \\ c_d^2(i-1) & i > 1 \end{cases} \tag{18.3}$$

where for $i = 1, \ldots, M$,

$$c_d^2(i) = 1 + [c_a^2(i-1) - 1][1 - u^2(m)] + \frac{u^2(m)}{\sqrt{m}}[c_e^2(i) - 1] \tag{18.4}$$

For a given equipment configuration (i.e., choice of technology option at each station) we use Equation (18.2) to compute CT(m) and check the total cycle time constraint. If it is violated, we must consider more capacity or a lower variability option. The trick is to change the configuration in the most cost-effective fashion.

Before this can be done, however, we must have a starting point that has sufficient capacity. We call this a **capacity-feasible** solution and give an example of how to find it below.

18.2.1 Example: A Minimum Cost, Capacity-Feasible Line

Consider a four-station line with a throughput target of two and one-half jobs per hour or 60 jobs per day (running three shifts per day). Suppose the SCV of arrivals to the line is equal to 1.0 (recall that we termed this the moderate-variability case in Part II). Thus, TH = 2.5 jobs per hour and $c_a^2 = 1.0$ for the first station. Set the target cycle time for the line at CT = 16. To begin, assume that only one type of machine is available at each station (although we are allowed to choose the number of machines to install at each station). Table 18.1 gives the data for the four stations.

First, we perform a capacity check to determine the minimum number of machines we need at each station. We do this by solving Equation (18.1) for the minimum value of m that keeps utilization below one, that is,

$$u(m) = \frac{r_a t_e}{m} \quad m < 1$$

or

$$m > r_a t_e$$

Table 18.1 Basic Data for a Line Design Problem

Station	Fixed Cost ($000)	Unit Cost ($000)	t_e (hours)	c_e^2
1	225	100	1.50	1.00
2	150	155	0.78	1.00
3	200	90	1.10	3.14
4	250	130	1.60	0.10

Table 18.2 The Minimum Cost, Capacity-Feasible Solution

Station	Number of Machines	Utilization	Cost ($000)
1	4	0.94	625
2	2	0.98	460
3	3	0.92	470
4	5	0.80	900
Total			2,455

For the first station,

$$r_a t_e = 2.5 \text{ jobs/hour } \times 1.5 \text{ hours} = 3.75$$

which indicates we require at least four machines. Table 18.2 summarizes the other machine requirements and their corresponding utilization.

Note that for station 4,

$$r_a t_e = 2.5 \text{ jobs/hour } \times 1.6 \text{ hours} = 4.00$$

However, this would yield a utilization of exactly 1.0. Since the utilization law of factory physics stated that utilization must always be *strictly less than* 1.0, we must assign five machines to station 4, thereby lowering the utilization to 0.80.

Note that the solution in Table 18.2 is the least-cost configuration that has sufficient capacity. This is called the **minimum cost, capacity-feasible (MCCF)** configuration and in this case costs $2,455,000.

It is easy to extend this analysis to find the MCCF configuration when there is more than one technology option at each station. For each station we determine how many machines of each option are required to meet the capacity target and choose the option with the smallest total cost. Doing this for each station will result in an MCCF configuration for the line.

18.2.2 Forcing Cycle Time Compliance

Once we have a capacity-feasible configuration, we then check the cycle time, using Equations (18.2) and (18.4).

Station 1:

$$\mathrm{CT}(4) = \left(\frac{1.0 + 1.0}{2}\right)\left(\frac{0.94^{\sqrt{2(4+1)}-1}}{4(1-0.94)}\right)1.5 + 1.5 = 6.72 \text{ hours}$$

$$c_d^2 = 1 + (1-1)(1-0.94^2) + \frac{0.94^2}{\sqrt{4}}(1-1) = 1.0$$

Station 2:

$$\mathrm{CT}(2) = \left(\frac{1.0 + 1.0}{2}\right)\left(\frac{0.98^{\sqrt{2(2+1)}-1}}{2(1-0.98)}\right)0.78 + 0.78 = 15.82 \text{ hours}$$

$$c_d^2 = 1 + (1-1)(1-0.98^2) + \frac{0.98^2}{\sqrt{2}}(1-1) = 1.0$$

Station 3:

$$\mathrm{CT}(3) = \left(\frac{1.0 + 3.14}{2}\right)\left(\frac{0.92^{\sqrt{2(3+1)}-1}}{3(1-0.92)}\right)1.1 + 1.1 = 8.87 \text{ hours}$$

$$c_d^2 = 1 + (1-1)(1-0.92^2) + \frac{0.92^2}{\sqrt{3}}(3.14-1) = 2.0$$

Station 4:

$$\mathrm{CT}(5) = \left(\frac{2.0 + 0.1}{2}\right)\left(\frac{0.80^{\sqrt{2(5+1)}-1}}{5(1-0.80)}\right)1.6 + 1.6 = 2.59 \text{ hours}$$

The sum of these cycle times is 34 hours, which is significantly greater than the target of 16. Clearly, the line needs changes to obtain a design that complies with the strategic specifications.

There are three basic improvement alternatives: (1) Modify the existing machines, (2) change the machine options, or (3) add more machines. Chapter 9 described how to use factory physics principles to *diagnose* problems in a line. This approach could be used to determine the cause of long cycle times (e.g., long and infrequent outages) and therefore what machine modifications would be most effective. It might be worthwhile to spend money to reduce variability or speed up a machine rather than to purchase an additional one. Of course, if we are designing a new line, there are no "existing" tools, and hence alternative one is not available.

Altering machine options in the pursuit of shorter cycle times might entail purchasing a different and perhaps more expensive machine with better operating characteristics (e.g., faster rate or smaller process variability). Often, however, especially in high-tech situations, the number of distinct machine types is quite limited. In some cases there may be only a single equipment vendor available. When this is the case, most of the technology options that can be used to reduce cycle time are modifications of a given machine type. Modifications include speeding up the machine, reducing setup time, reducing MTTR, and so on.

The most obvious way to reduce excess cycle time is simply to purchase more machines. If capacity comes in small increments, this might well be the most economical approach.

Depending on the size of the required reduction in cycle time, the range of available technology options, and the cost and size of capacity increments, the best approach may consist of any number of combinations of these types of alternatives.

18.3 Modifying Existing Production Lines

We now offer a heuristic procedure for determining a least-cost configuration that meets the throughput and cycle time constraints. The heuristic starts with the MCCF configuration and then looks for the change that results in the "biggest bang for the buck" with respect to cycle time improvement.

To illustrate this approach, we reconsider the example of Table 18.1. Recall that the minimum cost, capacity-feasible configuration (Table 18.2) did not satisfy the cycle time constraint. Specifically, desired total cycle time was 16 hours, but the resulting total cycle time of the minimum cost configuration was 34 hours. We now consider how to bring the configuration into cycle time compliance in a cost-efficient fashion. Note that this is precisely the type of problem faced by firms trying to implement the methods of cycle time reduction or time-based competition in an existing facility.

To make the example more realistic, suppose we can modify as well as add machines at each station. In particular, suppose that by spending $10,000 per machine at the third station, we could alter long and infrequent random outages to shorter but more frequent ones with the same availability (recall the discussion in Chapter 8 that showed why this is desirable). We might be able to accomplish this by installing field replacement parts and/or doing more preventive maintenance. We assume here that this does not change t_e, but does reduce c_e^2 from 3.14 to 1.0. Using these cost and performance data, we can consider this variability reduction option as an alternative to adding machines.

Hence, these are the available options: At any station, we can add a machine; at station 3, we can either add a machine or reduce machine variability by changing the characteristics of the machine. For each alternative, we can compute the change in cycle time at the station and the change in cost.[1] A reasonable measure of the effectiveness of the change is the ratio of the change in cost to the change in cycle time. The "best single change" is that with the lowest ratio. We compute these ratios for each option in Table 18.3.

The first thing we notice from Table 18.3 is that no single change reduces total cycle time by enough to satisfy the cycle time constraint—we need an 18-hour reduction. The smallest ratio is obtained by modifying the machine at station 3 (by reducing repair time variability) with cycle time reduced by 4.49 hours at a cost of $30,000. This takes us down to 29.51 hours, still considerably longer than the 16 hours allotted. If we repeat the analysis, the minimum ratio occurs by adding a machine to station 2, which costs $155,000 and further reduces cycle time by 14.7 hours. This takes us down to 14.81 hours, which is within the 16-hour constraint.

Although we are not guaranteed that repeatedly choosing the best single change will bring us within the cycle time constraint at a minimum cost, this approach usually works well. In any case, it does yield a configuration that is throughput- and cycle time–feasible. For this example, the resulting solution is given in Table 18.4.

The total cost is $2,640,000, or $185,000 more than the MCCF configuration. In addition, notice that this line is not even close to balanced. Surprisingly, the most expensive station (number 4) has the lowest utilization. This is because both the fixed cost and the unit cost at station 4 are quite high, and because four machines at station 4 result in 100 percent utilization.

[1] We ignore what might happen downstream at this point, so our calculations are actually approximations of the change in cycle time for the entire line. It is easy enough to go back and check the line cycle time for a specific option, and for that matter it is not too hard to include downstream effects when estimating the effect of a single change. However, if we do this, we can only evaluate changes one at a time—the reduction in total cycle time from two options together is *not* necessarily the sum of the reductions from each separately.

TABLE 18.3 Cost and Cycle Time Impacts of Improvement Alternatives

Station	Current Number of Machines	Change	Cost Increase ($000)	CT Decrease (hours)	Ratio ($000/hour)
1	4	Add machine	100	4.63	21.61
2	2	Add machine	155	14.73	10.52
3	3	Add machine	90	7.20	12.49
3	3	Reduce variability	30	4.49	6.67
4	5	Add machine	130	0.71	183.10

TABLE 18.4 Capacity- and Cycle Time–Feasible Configuration

Station	Number of Machines	Utilization	Station Cost ($000)
1	4	0.94	625
2	3	0.65	615
3	3 (modified)	0.92	500
4	5	0.80	900
Total			2,640

18.4 Designing New Production Lines

The problem of designing a new line is different from that of modifying an existing one, in that there are typically many more options to consider. In a new line, we are not constrained by existing machines, facilities, or even structure. Indeed, we may have *so much* freedom that the problem becomes almost impossible to solve in an optimal fashion.

18.4.1 The Traditional Approach

In the 18th century, when the first factories were designed, a major consideration was how to arrange the various operations in order to run them from a single source of power—the waterwheel. Consequently, operations were arranged in linear fashion along the waterwheel shaft, each connected to a belt on a properly sized gear to obtain the required turning speed from the waterwheel. Today, it is not uncommon to find factories that follow this traditional design, their process centers laid out in straight lines within a rectangular facility.

We found this curious, since manufacturing plants have not relied on water power for 150 years, and we questioned several architectural engineers who design complex plants (e.g., wafer fabs) and manufacturing engineers who work in existing plants. We discerned that a typical procedure for designing new plants and new lines goes something like this:

1. Establish the basic size and shape of the new facility.
2. Determine where the support facilities (electricity, steam headers, process gases, etc.) should go to minimize the cost of the facility.

3. Determine where the workstations should go within the facility to minimize cost.
4. Determine the product flow.

Given this, the tendency toward linear layouts is not surprising. Since the design process *starts* with the size and shape of the facility, tradition exerts strong influence over the resulting design. But there are obvious problems with this scheme. The most serious is that little consideration is given to product flow until after most of the plant has been designed.

18.4.2 A Factory Physics Approach

A good alternate approach is to view the problem from a customer perspective. This makes it clear that the main purpose of a line or plant is to provide quality product in a timely and competitive fashion. A facility design process consistent with this goal, which is almost the reverse of the traditional approach, is the following:

1. The customer determines the product. Mixes, volumes, and cycle times are forecast.
2. The product(s) determine(s) the processes. For most products, there is a basic recipe of steps that must be done to produce a unit.
3. The processes determine a basic set of machines. Machine descriptions will start out very general and will acquire detail as the planning process evolves.
4. The machines determine the facilities needed to support them.
5. The facilities determine the overall structure and size of the plant.

Of course, if we were to literally follow this procedure, we could end up with a facility that is well equipped to make the product in the volumes desired but is too costly to build. Focusing solely on product flow in order to minimize cycle times may lead us to install multiple expensive machines when one would have done. For instance, in a wafer fab, the photolithography operation is typically one of the more expensive machines in the fab. Its facility requirements are enormous, and to make matters worse, the wafers must visit the operation for each layer (often 10 or more) applied during fabrication. A pure cycle time minimization perspective might suggest installing 10 sets of equipment at a tremendous cost. A pure cost minimization perspective would call for only one set of equipment. The "best" option can only be determined by considering photolithography in the context of the other operations and comparing relative costs of different configurations that meet performance targets.

As a result, it makes sense to approach the facility design problem from a combination of the traditional and factory physics perspectives. We start with an idea of the basic processes and layout of the factory. Using the basic layout, we install the process centers, sizing them to meet desired throughput and cycle time levels. If the resulting configuration results in too high a facility cost, we reconsider the basic layout. On the other hand, if cycle times are excessive, we consider installing more support facilities to improve process flows.

As part of the analysis, we might also want to do a Pareto analysis of the product mix to determine if a "factory within a factory" concept is applicable. If most of the volume is for a relatively small number of products, it may make sense to duplicate processes in the plant. One set, in a tight flow line configuration, is dedicated to the small number of products representing the large portion of throughput. The other is arranged in more of a job shop configuration that maximizes flexibility at the expense of lower utilization or

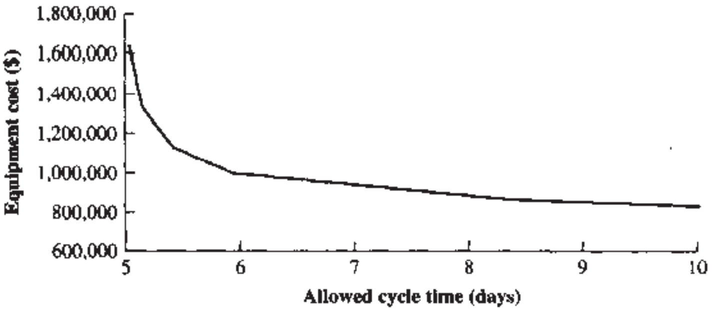


FIGURE 18.2

Plot of total equipment cost versus total cycle time

higher cycle times. Low utilization should be expected in this portion since the volumes are (by design) low.

Once we have settled on a basic layout, we turn to detailed selection of specific options and numbers of machines. A relatively simple procedure is to start with the MCCF configuration and then successively choose the best single change, as described, to bring the line into cycle time compliance. To be effective, we should include as many available technology options (i.e., including both purchasing additional machines and modifying machines and/or procedures on site) as we can without overwhelming the decision maker. We want to avoid overlooking an inexpensive modification that alleviates a performance problem and eliminates the need for additional expensive machines. Factory physics diagnostic procedures (Chapter 9) are useful in identifying promising options.

Of course, as we know, the performance requirements (e.g., throughput and cycle time targets) are themselves decision variables. Although we can specify plausible values to start the analysis, it makes sense to examine tradeoffs between cost and performance. For example, if we could shorten cycle times by five days at a cost of $100,000, we might well decide to do it. We can do this with our model by solving it for various values of the throughput or cycle time constraints in order to generate a cost-versus-performance curve. A typical plot of cost versus total cycle time is shown in Figure 18.2. While the model cannot specify which point on this curve is optimal, it does provide useful information to help the decision maker make a rational choice.

18.4.3 Other Facility Design Considerations

These discussions offer some perspective on how to incorporate cost, throughput, cycle time, and other factors into a customer-oriented facility design process. However, there is more to the facility design problem than we have dealt with here. Indeed, there exists a vast literature called, broadly, **plant layout** or **facilities planning,** which deals with topics ranging from the placement of various process centers to minimize product flow, to determining the number of employee parking spaces. This literature addresses the important issues of materials handling, physical plant layout, storage and warehousing, office planning, facility services, and developing and maintaining facilities plans. We suggest Tompkins and White (1984) as a good introduction to this field.

18.5 Capacity Allocation and Line Balancing

As the previous example illustrated, factory physics procedures for line design are unlikely to result in a balanced line. The reasons are as follows:

1. An unbalanced flow line with distinct bottleneck is easier to manage and exhibits better logistical behavior (i.e., has a characteristic curve closer to the best case) than a corresponding balanced line.
2. The cost of capacity is typically not the same at each station, so it is cheaper to maintain excess capacity at some stations than at others.
3. Capacity is frequently available only in discrete-size increments (e.g., we can buy one or two lathes, but not one and one-half), so it may be impossible to match capacity of a given station to a particular target.

When appropriate consideration is given to these factors, the optimal configuration of most flow lines will be an unbalanced line.

18.5.1 Paced Assembly Lines

Despite the arguments in favor of unbalanced lines, sometimes line balancing makes sense. Indeed, the line-of-balance (LOB) problem is a classic problem in industrial engineering. However, it is applicable only to **paced assembly lines,** not **flow lines.** In a flow line, stations are essentially independent. Each station operates at its own speed, so the bottleneck is the slowest station in the line. In a paced assembly line, parts flow through the line on a belt or chain that moves at a constant speed. The parts move through **zones** that usually contain one or more operators. The line is designed so that the operators will almost always be able to complete their task while the part is in their zone. If not, the line would be disrupted as workers tried to finish tasks in the next worker's zone. Hence, the bottleneck of a paced assembly line is not the slowest station in the line but the line-moving mechanism itself.

Additionally, capacity increments in a paced assembly line are usually much smaller than those in a flow line. In a paced assembly line, tasks are typically assigned to workers on the line and can be split into fine increments. For example, in a manual electronic assembly operation, each station "stuffs" circuit boards with a number of components. Since there are many components, the line can be balanced by adjusting the amount of stuffing done at each station. A discussion and an example technique for solving the LOB problem are given in Appendix 18A.

Another justification for a balanced assembly line is one of personnel management. No one likes to be in a situation in which they are constantly expected to do more than their peers for the same pay. Since most assembly lines are staffed by people (although some assembly lines use robots), the issue of fairness is an important one. In these cases a line in, which each station has nearly the same amount of work is desirable.

In contrast, in a flow line, the tasks depend more on the machines themselves and are therefore less easily divided. To increase capacity at a particular station, we must either add an additional machine to that station or speed up the existing ones. Unfortunately, the notion of a balanced line has become so ingrained that it is often applied when it is inappropriate. This and the desire to have high utilization are the reasons one frequently encounters nearly balanced flow lines.

18.5.2 Unbalancing Flow Lines

The previous reasons for unbalancing flow lines suggest that a process with small and inexpensive capacity increments should never be a bottleneck. Such a process can easily and inexpensively add small increments of capacity until it no longer causes problems due to insufficient capacity. On the other hand, a process for which capacity comes in large expensive blocks is a good choice to be the line bottleneck.

As an example, consider two different process centers in a circuit board plant: copper plate and manual inspect. The manual inspect operation occurs before the copper plate operation.[2] Copper plate utilizes a machine that involves a chemical bath along with enormous amounts of electricity. Each machine has a capacity of around 2,000 panels per day. Adding an additional machine at copper plate costs more than $2 million in machine and facility costs and requires a significant amount of floor space. Copper plate represents one of the largest and most expensive machines in the plant. In contrast, each of the stations in manual inspect requires one semiskilled operator, an illuminated magnifier, and a touch-up tool. Each station can inspect around 150 panels per day. None of these stations costs more than $100, and the floor space requirements are small.

If these were the only two stations in the line, the situation would be easy to analyze. If we designate the copper plater to be the bottleneck, then we can easily and inexpensively keep it from starving by adding capacity to the manual inspect operation. It is of little consequence that manual inspect is not fully utilized. On the contrary, to designate manual inspect as the bottleneck and to keep it from starving,[3] we would have to add a large and costly increment of capacity to the copper plate operation. Thus, it makes more sense to designate copper plate as the bottleneck and to manage it accordingly.

18.6 Conclusions

This chapter has focused primarily on applying the factory physics framework to the design of new production lines and improvement of existing ones with respect to capacity. Our main points can be summarized as follows:

1. *Capacity decisions have a strategic impact on the competitiveness of the manufacturing operation.* A capacity strategy has a strong direct effect on costs and many indirect effects on performance by influencing other planning and control problems, including aggregate planning, scheduling, and shop floor control. Decisions include how much, when, where, and what type of capacity to add. Other strategic issues involve various economies and diseconomies of scale.

2. *Factory physics formulas can provide the basis for line design and improvement procedures.* By allowing computation of throughput, cycle time, and WIP for a given configuration, these formulas enable us to frame the line design or improvement problem as one to minimize cost subject to specified throughput, cycle time, and/or WIP constraints. By varying the constraints, we can also generate cost-versus-performance constraints.

3. *Capacity additions and equipment or procedure modifications can be viable alternatives and/or complements to one another.* For instance, reducing repair times on an existing machine can sometimes have similar logistical effects as adding capacity to a station in the form of additional machines. All other things being equal, the value of procedural changes is typically greater than that of equipment additions, because the learning and discipline gained from improving a line can be translated to other lines, while simple capacity additions offer no such learning opportunities.

[2] The capacities, capabilities, and even the process description have been altered here from those in a circuit board plant in which the authors have consulted.

[3] Recall that in a CONWIP line, there really is no *front* to the line. Thus, workstations earlier in the line can be starved by later workstations if the pull signals (i.e., the CONWIP "cards") are not returned in a timely manner.

4. *Flow lines should generally be unbalanced.* Logistical and cost differences between stations make it sensible to configure flow lines to have different levels of utilization at the stations.

5. *Paced assembly lines should generally be balanced.* On paced assembly lines it is the pacing mechanism (e.g., the conveyor or chain) that is typically the bottleneck. To enable workers to complete their assigned tasks within the allotted pacing time, as well as to allocate work fairly, it makes sense to divide tasks among stations as evenly as possible, subject to precedence and discreteness requirements.

It is important to note that lines designed using factory physics procedures are likely to be more expensive than lines designed using a traditional minimum cost, capacity-feasible approach. However, they are also much more likely to do what they were designed to do. When one considers factors such as lost sales due to inability to meet throughput targets, loss of customer goodwill due to inability to meet cycle time targets, and the confusion that results in trying to operate a line that is in a constant state of chaos, the more expensive factory physics lines are likely to be much more profitable in the long run.

Appendix 18A
The Line-of-Balance Problem

Assigning tasks to stations on a paced assembly line should be done so that each station has nearly the same amount of work. There are two good reasons for this: to use labor efficiently and to avoid issues of fairness that result when one station must work much harder than another.

Assume there are n tasks to be performed on each piece moving through the line and the time to do the ith task is t_i. These tasks are assigned to k workstations where $k \leq n$. If t_0 is the time allowed for each station (i.e., the time for the conveyor to move through a workstation), then the rate of the line will be $r_b = 1/t_0$.

Since the tasks have random times, we need to make some allowance for variability. We define $c < t_0$ to be the maximum time allowed for task assignment. By requiring the sum of the mean task times to be less than or equal to c, we provide some extra time at each station to accommodate the inherent variability of the tasks. Note that $u = c/t_0$ is the maximum utilization of any station in the line and is always less than one.

In many texts dealing with the LOB problem, c is called the *cycle time*. However, since we use this term to refer to the time through an entire routing, we will refer to c as the **conveyor time** (i.e., because it is the time the conveyor allows at each station).

The objective of most line-of-balance algorithms is to minimize total idle time, which we write as

$$\text{Total idle time} = kc - \sum_{i=1}^{n} t_i$$

An equivalent measure is known as **balance delay**

$$b = \frac{kc - \sum_{i=1}^{n} t_i}{kc}$$

which represents the total fraction of idle time.

To further complicate matters, we must consider a number of other constraints. The most common are **precedence** constraints, which occur when certain tasks must be done before others. We will consider only precedence constraints, but refer the reader to Hax and Candea (1984, section 5.4) for a more complete discussion of the LOB problem and a survey of relevant literature.

It turns out that the LOB problem is very complex (i.e., NP-hard), so that optimal algorithms often require excessive amounts of computer time for realistically sized problems (e.g., with 100 tasks or more). For this reason, most commercial packages rely on heuristic methods.

We illustrate a heuristic LOB algorithm using a simple procedure that is similar to that of Kilbridge and Wester (1961) by using an example from Johnson and Montgomery (1974, p. 369). To do this, consider the nine tasks whose precedence relations are given in Figure 18.3. The times for these tasks and the number of successors are given in Table 18.5. Note that task 5 has the largest average performance time of 10. Thus, $c \geq 10$. Also note that the sum of the performance times is $\sum_i t_i = 48$.

To have zero idle time, the ratio $\sum_{i=1}^{n} t_i/c$ must be an integer. However, this does not guarantee zero idle time because the precedence constraints might prevent the required assignment of tasks to stations. Nonetheless, this fact and

$$\max_i(t_i) \leq c \leq \sum_{i=1}^{n} t_i$$

help to determine an appropriate value for c. If we factor $\sum_{i=1}^{n} t_i = 48$, we get

$$2 \times 2 \times 2 \times 2 \times 3 = 48$$

The combinations of these factors that are between 10 (the largest performance time) and 48 (the sum of the performance times) are

$$\begin{aligned} 2 \times 2 \times 2 \times 2 \times 3 &= 48 \\ 2 \times 2 \times 2 \times 3 &= 24 \\ 2 \times 2 \times 2 \times 2 &= 16 \\ 2 \times 2 \times 3 &= 12 \end{aligned}$$

FIGURE 18.3

Precedence diagram for LOB example

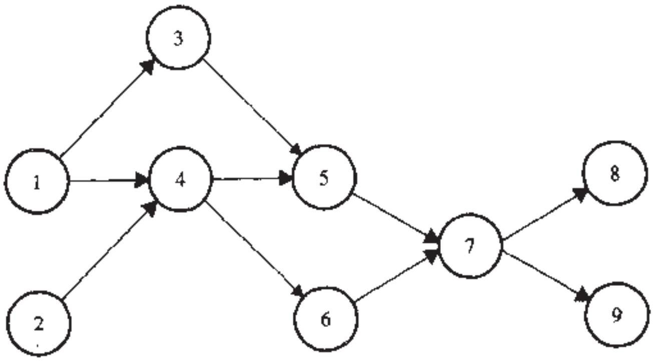


TABLE 18.5 Data for LOB Problem Example

Task Number	Average Performance Time	Number of Successors
1	5	7
2	3	6
3	6	4
4	8	5
5	10	3
6	7	3
7	1	2
8	5	0
9	3	0

So we *might* be able to achieve a perfectly balanced line (i.e., no idle time) with either $48/48 = 1$ station (obvious and not very useful), $48/24 = 2$ stations, $48/16 = 3$ stations, or $48/12 = 4$ stations. Let us consider the case with $c = 16$, the three-station case.[4]

To describe our procedure, define N to be the current station number, T the set of tasks assigned to the current station, A the time available to be assigned at the current station, and S the set of available tasks to be assigned, that is, those tasks whose precedence constraints have been satisfied and whose performance times fit within the remaining time. The algorithm then proceeds as follows:

Step 1. Set the current station number N to 1.

Step 2. Set the time available to c, $A \leftarrow c$, and $T = \phi$, indicating no assignments thus far.

Step 3. Determine the set of candidate tasks for assignment S. To be a candidate, two conditions must be satisfied:

1. All predecessors of the candidate must be scheduled, or equivalently, the candidate has no predecessors.
2. The performance time does not exceed the time available: $t_j \leq A$.

Step 4. Choose the task j from the set S, using the following two rules:

1. Choose the task that has the largest number of total successors.
2. Break ties by choosing the task with the longest performance time.

Place the task in T.

Step 5. Update the available time $A \leftarrow A - t_j$. Remove task j from set S.

Step 6. Repeat steps 3, 4, and 5 until no candidate tasks remain (i.e., set S is empty).

Step 7. If there are tasks remaining, increment the station number and go to step 2. Otherwise, stop.

To apply this algorithm to our example, we start with

$$N = 1 \qquad A = 16 \qquad S = \{1, 2\} \qquad T = \phi$$

Set S contains tasks 1 and 2 only, since they are the only tasks without any predecessors. Since task 1 has the most successors, we assign it first to station 1. We now have

$$N = 1 \qquad A = 11 \qquad S = \{2, 3\} \qquad T = \{1\}$$

Note that task 3 is now a candidate since its only precedence, task 1, has been scheduled. Since task 2 has the most successors and fits within the available time, we schedule it next.

$$N = 1 \qquad A = 8 \qquad S = \{3, 4\} \qquad T = \{1, 2\}$$

Both tasks 3 and 4 are now candidates for the next slot. Here we see the importance (and arbitrariness) of the heuristic rules. Since our rule is to select the task with the most successors, we select task 4 which fits perfectly (using all eight time units remaining). If we had selected task 3, we would have had time remaining at the station after the task assignments. More sophisticated LOB algorithms would try all combinations of the tasks remaining and see if any are a perfect fit. This, of course, increases the amount of computer time required. The status of the algorithm is now

$$N = 1 \qquad A = 0 \qquad S = \phi \qquad T = \{1, 2, 4\}$$

There are no candidate tasks because the time remaining is zero. We must now move on to schedule the second station. We reset $A = c$ and note that there are now two candidate tasks

$$N = 2 \qquad A = 16 \qquad S = \{3, 6\} \qquad T = \phi$$

Task 3 has the greatest number of successors and so is scheduled first at station 2. The status is now

$$N = 2 \qquad A = 10 \qquad S = \{5, 6\} \qquad T = \{3\}$$

[4] Of course, by choosing the value $c = 16$ we have established the throughput of the line. If we need greater throughput, we might be better off with $c = 12$, even though the line will not be perfectly balanced and even though there is more idle time. These issues are often not considered in LOB software.

Tasks 5 and 6 both have three successors. However, task 5 is the longest task and just fits in the time remaining. We finish station 2 with

$$N = 2 \quad A = 0 \quad S = \{6\} \quad T = \{3, 5\}$$

The remaining tasks all fit within the conveyor time c at station 3.

$$N = 3 \quad A = 0 \quad S = \phi \quad T = \{6, 7, 8, 9\}$$

The schedule is optimal with $b = 0$.

Note how many times during the algorithm that we got lucky when tasks "just fit" in the time remaining. This is not typical and, in fact, would not happen when $c = 12$ or $c = 24$. Most commercial algorithms try many different values of c and different tie-breaking rules within the procedure.

Study Questions

1. Why would anyone want to add capacity before demand has materialized? Why would anyone want to lag behind demand?
2. Why is the unit cost usually less expensive in a large plant than in a small one? What might cause this not to be true?
3. Why is the traditional view of capacity management inadequate? What law from factory physics speaks to this directly?
4. Consider this statement: For a fixed budget, design the "best" facility possible. Provide a more specific problem statement in terms of cost, cycle time, throughput, and so on.
5. Why is it appropriate to balance a paced assembly line but not a line of independent workstations? What is the bottleneck of a paced assembly line?
6. Consider the line-of-balance problem. Why should the conveyor time c be greater than the maximum time assigned at any station? What might happen if it were not?
7. What are some shortcomings of the traditional approach to designing factories in which we start with the size and shape of the plant, decide where the support facilities go, and then decide where to place the tools? What are some shortcomings of the factory physics approach?

Problems

1. You are charged with designing a three-station flow line that must achieve a target throughput of five jobs per hour and a total cycle time of three hours or less. Each station must consist of a single machine purchased from a vendor who will construct it to your specifications, any speed you desire. However, the price depends on the speed as follows:

$$K(i) = a(i)\left[\frac{1}{t_e(i)}\right]^{b(i)}$$

where $K(i)$ is the (total) equipment cost at station i; $t_e(i)$ is the effective process time of the machine at station i; and $a(i)$ and $b(i)$ are constants. Assume that the arrival coefficient of variation (CV) to the line is equal to one and that $c_e(i) = 1$ for $i = 1, 2, 3$ (i.e., the process CV for all machines is equal to one, regardless of the speed).

a. Suppose that $a(i) = \$10,000$ and $b(i) = \frac{2}{3}$ for $i = 1, 2, 3$. Find the values of $t_e(i)$ for $i = 1, 2, 3$ that achieve target throughput and cycle time with minimum total equipment cost. (*Hint:* The *Solver* tool in Excel is very handy for this.) Is the result a balanced line? Explain why or why not.

b. Suppose that $a(1) = \$1,000$, $a(2) = \$100,000$, $a(3) = \$10,000$, and $b(i) = \frac{2}{3}$ for $i = 1, 2, 3$. Find the values of $t_e(i)$ for $i = 1, 2, 3$ that achieve target throughput and cycle time with minimum total equipment cost. Is the result a balanced line? Explain why or why not.

c. Suppose that everything is the same as in part *a* except that now $t_e(i)$ can only be chosen in multiples of 0.05 hour (0.05, 0.1, 0.15, etc.). Find the values of $t_e(i)$ for $i = 1, 2, 3$ that achieve target throughput and cycle time with minimum total equipment cost. Is the result a balanced line? Explain why or why not.

d. What implications do the results of this simplified model have for designing realistic flow lines?

2. Table 18.6 gives the speeds (in pieces per hour), the CV, and the cost for a set of machines for a circuit board line. Jobs go through the line in totes that hold 50 panels each; this cannot be changed. The CVs represent the *effective* process times and thus include the effects of downtime, setups, and other common disruptions.

   The desired average cycle time through this workstation is one day. The maximum demand is 1,000 panels per day.

   a. What is the least-cost configuration that meets demand requirements?

   b. How many possible configurations are there?

   c. Find a good configuration.

3. *Challenge:* Consider the data in Table 18.1 along with the option of reducing the c_e^2 for station 3 as described in Section 18.3. Design a line with maximum throughput that has cycle times of not more than 16 hours and an equipment budget of no more than $2,800,000.

4. Assembling a computer monitor requires a chassis, two main circuit boards and components, a yoke, followed by a test. These are performed according to the following precedence requirements:

   - The chassis must be put down first. This takes two minutes.
   - Board 1 requires only a chassis. It takes three minutes.
   - Components 1 require that board 1 be in place. Placing these components on the board takes three minutes.
   - Board 2 requires that board 1 be in place. Board 2 takes four minutes to insert.
   - Components 2 require that board 2 be in place. These take two minutes to insert.
   - The yoke requires that all the boards and the components be in place and takes three minutes to install.
   - Testing, naturally, requires that all the assembly be finished and takes five minutes to perform.

   a. Draw a precedence diagram of the assembly of a computer monitor.

   b. What is the minimum conveyor time that could possibly result in zero balance delay?

   c. If the expected utilization is 0.85, how many monitors will be produced per hour using the minimum conveyor time computed above?

   d. Assign the tasks to stations using the minimum conveyor time. What is the balance delay?

Table 18.6 Possible Machines to Purchase for Each Work Center

	Possible Machines (Speed (pieces/hour), CV, Cost ($000))			
Station	**Type 1**	**Type 2**	**Type 3**	**Type 4**
MMOD	42, 2.0, $50	42, 1.0, $85	50, 2.0, $65	10, 2.0, $110.5
SIP	42, 2.0, $50	42, 1.0, $85	50, 2.0, $65	10, 2.0, $110.5
ROBOT	25, 1.0, $100	25, 0.7, $120	—	—
HDBLD	50, 0.75, $20	5.5, 0.75, $22	6, 0.75, $24	—

CHAPTER

19 Synthesis—Pulling It All Together

This is not the end. It is not even the beginning of the end. But it is, perhaps, the end of the beginning.

Winston Churchill, November 10, 1942

19.1 The Strategic Importance of Details

We will be the first to admit that the treatment of manufacturing in this book has been technical. Manufacturing *is* technical. It would be nice if we could just do what feels right, get product out the door, and make a living. But there are fewer and fewer businesses in which this is possible. Under the pressure of intense global competition, manufacturing firms are *forced* to continually improve cost efficiency, product quality, and delivery responsiveness. Certainly a strategic vision is essential to foster an environment where this kind of performance is possible. But it is only through careful attention to technical detail that it can be achieved.

In the 1950s and 1960s America could afford to gloss over the details of manufacturing and concentrate on high-level marketing and finance issues. In the wake of World War II, American manufacturers did not need to worry about costs or defect levels that were a few percent too high. Customers had few alternatives and low expectations. In the 1980s and 1990s, however, consumers began to see high-quality, reasonably priced products from Japan, Germany, Korea, and many other places, and accordingly, they grew to expect more from American manufacturers. As a result, today even a relatively small gap in cost, quality, or customer service can drive a firm right out of a market.

The strategic value of details, however, goes well beyond their role in achieving small but important performance improvements. The most important reason that we need a deeper understanding of manufacturing systems is that the pace of technological change in recent years has made trial-and-error solutions almost useless. Henry Ford produced the Model T for an entire generation, so he could evolve systems and solutions by observing and tinkering with the production line. In contrast, the typical life span of a personal computer is less than two years, which means that modern PC manufacturers must set up the facilities, ramp up the volumes, attain the efficiencies needed to make a profit, achieve the level of predictability needed to ensure good customer service, and phase out the product, all in a very short time. Predicting and analyzing the behavior of a system *before* it is in place requires sound intuition and appropriate models, both of which are premised on an understanding of the technical details of manufacturing.

19.2 The Practical Matter of Implementation

Having the proper analysis tools is a key prerequisite for making significant improvements to a manufacturing system. But implementation is more than a matter of being right. An effective manufacturing manager must pull together a coherent plan and nurture it to fruition. This requires (1) addressing the *right* problem and (2) convincing others that it needs to be solved. The first is the subject of systems analysis, while the second deals with the human element of manufacturing management. Chapters 6 and 11 addressed these; but they are so central to the implementation process that we revisit them briefly here.

19.2.1 A Systems Perspective

The laws and formulas of factory physics can help identify areas of leverage, build intuition about why certain approaches work in certain environments, and evaluate and compare specific policies. But they cannot generate original ideas. The managers of a manufacturing system must determine *what* they want it to do before any tools can be applied to the question of *how* to do it. Therefore, to fully exploit the strategic potential of factory physics, it is important to use it in the larger problem-solving framework of systems analysis.

Recall from Chapter 6 that the essential aspects of systems analysis (as well as the modern variant of systems analysis, business process reengineering) are as follows:

1. *A systems view.* The problem is viewed in the context of a system of interacting subsystems. The emphasis is on taking a broad, holistic view of the problem, rather than a narrow, reductionist one.

2. *Means-ends analysis.* The objective is always specified first, and then alternatives are sought and evaluated in terms of this objective. For instance, a systems analysis project might use the objective "to deliver finished goods swiftly and conveniently to customers," but would *not* use the objective "to improve the efficiency of processing purchase orders." The latter is a "means-first" approach, which could rule out potentially attractive options—such as doing away with purchase orders under an entirely new procedure.

In systems analysis, objectives are typically organized into a hierarchy of objectives, which identifies the links between the fundamental objective and various lower-level objectives. This helps identify conflicting objectives (e.g., low inventory and high fill rate) and highlights lower-level objectives that support more than one higher-level objective (e.g., short cycle times allow for better manufacturing quality as well as better customer responsiveness).

3. *Creative alternative generation.* With the objective in mind, the systems approach seeks as broad a range of alternate policies as possible. For instance, to reduce manufacturing cycle time, we should go beyond simply considering how to speed up individual processes and think about basic causes of cycle time. Many formalized brainstorming techniques have been developed to encourage expansive thinking about nonobvious alternatives.

4. *Modeling and optimization.* To compare alternatives in terms of the objective, the project requires some kind of quantification. The modeling/optimization step for doing this may be as simple as computing costs for each alternative and choosing the cheapest one, or it may require analysis of a sophisticated mathematical model. The appropriate level of detail will vary depending on the complexity of the system and the magnitude of the potential impact.

5. *Iteration.* In every complex systems analysis project, the objective, alternatives, and model are revised repeatedly. This is because as we perform the analysis, we learn more about the system. In Chapter 6, we formalized this procedure as the "conjecture and refutation" process.

The systems analysis procedure helps focus attention on the correct problem (i.e., where major leverage exists), promotes insight into the system, and fosters a sense of teamwork toward the project. As such, it is a vital starting point and frame of reference for virtually any manufacturing improvement project.

19.2.2 Initiating Change

Systems analysis is valuable in generating and evaluating ideas. But no matter how good an idea is, it will never be implemented if it cannot be communicated. All the factory physics arguments in the world will not change a manufacturing organization unless the people in it are convinced of the need for change and know what they must do to bring it about.

Overcoming institutional momentum can be very difficult. As Machiavelli put it:

> There is nothing more difficult to take in hand, more perilous to conduct, or more uncertain in its success, than to take the lead in the introduction of a new order of things.

The amount of effort required to put through a program of change depends on the situation. If the manager of a production line has used her factory physics insight to recognize that reducing setups on a particular machine would reduce WIP and cycle time, and she has the authority to form a setup reduction team consisting of machine operators and staff engineers, then she should probably go ahead and do it. No hoopla, slogans, or revolutions are required to make small, incremental changes in the system. And while such changes will not remake the company, they can be important parts in the process of ongoing improvement.

Bigger changes, such as refocusing a plant as part of a time-based competition strategy, require much more institutional support. Radically reducing customer lead times by addressing the entire product delivery process—which involves sales, order entry, manufacturing, customer service, and possibly many other functions—demands the leadership of someone with sufficient clout to make the necessary changes. Depending on the system, this might be the plant manager, or if influence beyond the plant is needed (e.g., product development or component production), someone even higher, perhaps the vice president for manufacturing or chief operating officer. Once the leader has been assigned, it is critical for him/her to instigate the change *and* provide ongoing support for it. If the leader gives a few fiery speeches and then disappears, momentum for change will quickly evaporate.

An effective leader with the requisite authority can get people inspired to change, but cannot actually carry out the change. Systems analysis teams are typically needed to do the analysis and oversee the implementation required to actually reshape an organization. These teams can be configured and managed in many different ways (see Hayes, Wheelwright, and Clark 1988; Hammer and Champy 1993 for examples). We will not go into a great deal of depth about this, but we make the following observations about systems analysis teams:

1. Teams should *not* be committees. That is, they should be small enough to function aggressively. If the number of people on a team exceeds 10 or so, it becomes so difficult to get everyone together that the team becomes ineffective.

2. The team should consist of key people from the major functional areas affected by the change. For instance, a cycle time reduction effort should involve people from sales, manufacturing, production control, and so on. These people must be chosen to have a "big picture" attitude, so that they are not simply protecting their turf. Alternatively, they could be assigned 100 percent to the systems analysis team with the knowledge that after the team is dissolved, they will *not* go back to their previous position. The idea is to motivate people to think in terms of what is good for the overall system, not just for their part of it.

3. The team should include some outsiders, people not directly connected with the system under consideration. These çould be people from elsewhere in the organization or independent consultants. The purpose of these outsiders is to act as provocateurs who will challenge assumptions and traditions. It is altogether too easy for a team of all insiders to mistake the way things are for the way things must be.

When supported by an influential leader and well-chosen analysis team, a systems analysis can be a powerful tool for bringing about dramatic change in an organization.

19.3 Focusing Teamwork

Often in modern manufacturing organizations, it is not the big failures that are most damaging, but rather the small successes. A highly visible failure that occurs when a firm attempts to push out the envelope of manufacturing practice is a noble effort and a valuable learning opportunity. In the right environment (one that does not punish people for taking good risks or become overly conservative in reaction to a failure), such failures are necessary and positive steps on the road of continual improvement.

In contrast, small safe projects that make tiny improvements can ensure their leaders of positive performance evaluations, but can steadily undermine the competitiveness of a firm. The reason is that they sap the resources of the organization. A firm that devotes too much energy to the easy marginal improvements is open prey to a competitor who aims higher. In this era of intense competition, the "all safe" strategy is almost a sure formula for failure.

This observation implies that a critical first step in setting up a systems analysis team is to focus the team on a problem of real importance. One way to do this is to make sure the original topic of a systems analysis study is sufficiently broad to allow the team to identify the major areas of leverage for themselves. As illustration we offer the example of a systems analysis in which the authors participated some years ago. At the inaugural workshop, the objective was stated as increasing the efficiency of the painting process. After listening to a great many details about the problems in painting, we asked about the motive for improving painting and learned that manufacturing cycle times were too long relative to the competition. But after we asked more questions, we were able to estimate that painting accounted for less than one day of a 10-week cycle time. Eventually, we discovered that the single major determinant of cycle time was the order entry process, which accounted for four weeks or more. Thus, although we eventually arrived at an appropriate focus for the study, we would have gotten there much more efficiently had the initial focus been on something broad like "remaining profitable in the face of faster competition," instead of the restrictive "improving painting efficiency."

19.3.1 Pareto's Law

A basic tool for sifting through a complex manufacturing system and picking out the most important aspects is **Pareto's law,** also known as the 80-20 rule. Pareto originally offered it as the law of economics that 80 percent of the wealth is owned by 20 percent of the people. Applied more generally, it states that a large fraction of any problem (or benefit) is caused by a small fraction of the constituents. For instance, a small percentage of part numbers accounts for the majority of demand, a small number of maintenance items accounts for the majority of the maintenance budget, a small number of customers accounts for both a large fraction of sales as well as complaints.

Pareto's law can be used as a management guide, suggesting the "important few" be given separate treatment from the "less important many." The few high-volume part numbers might be dedicated to efficient flow lines, while the many lower-volume part numbers are produced in a less efficient job shop environment. The few high-volume materials might be delivered in daily just-in-time fashion, while the many low-volume materials are purchased and stocked in bulk. The few machines accounting for a large fraction of downtime may have dedicated repair kits and specialized procedures, while the many machines causing less downtime are handled with routine maintenance procedures. The few big customers might be (probably will be) given preferential treatment relative to the many small customers. In each case, the idea is to allocate limited resources to the places where they will do the most good.

Pareto's law can also be used as a simplification tool. For instance, the routings in a manufacturing plant may seem like a hopelessly intricate mess when all part numbers are considered. But when only major families are considered, a much simpler pattern may emerge. Studying this simplified system is likely to be tractable and to lead to an understanding of the essential behavior of the overall system.

19.3.2 Factory Physics Laws

Once the system has been pared down to a manageable level using Pareto's law, the fundamental tools at the disposal of a systems analysis team are the laws of factory physics. First and foremost, these offer intuition about the way a manufacturing system will tend to behave. Additionally, they provide analytical methods that can be supplemented by many other modeling and analysis techniques as appropriate to the particular study.

The following is a summary of the key factory physics principles that have been introduced in this book.

Law (Little's Law):

$$\mathrm{WIP} = \mathrm{TH} \times \mathrm{CT}$$

Law (Best-Case Performance): *The minimum cycle time for a given WIP level w is given by*

$$\mathrm{CT}_{\text{best}} = \begin{cases} T_0 & \text{if } w \le W_0 \\ \dfrac{w}{r_b} & \text{otherwise} \end{cases}$$

The maximum throughput for a given WIP level w is given by

$$\mathrm{TH}_{\mathrm{best}} = \begin{cases} \dfrac{w}{T_0} & \text{if } w \leq W_0 \\ r_b & \text{otherwise} \end{cases}$$

Law (Worst-Case Performance): *The worst-case cycle time for a given WIP level w is given by*

$$\mathrm{CT}_{\mathrm{worst}} = wT_0$$

The worst-case throughput for a given WIP level w is given by

$$\mathrm{TH}_{\mathrm{worst}} = \frac{1}{T_0}$$

Definition (Practical Worst-Case Performance): *The practical worst-case (PWC) cycle time for a given WIP level w is given by*

$$\mathrm{CT}_{\mathrm{PWC}} = T_0 + \frac{w-1}{r_b}$$

The PWC throughput for a given WIP level w is given by

$$\mathrm{TH}_{\mathrm{PWC}} = \frac{w}{W_0 + w - 1} r_b$$

Law (Labor Capacity): *The maximum capacity of a line staffed by n cross-trained operators with identical work rates is*

$$\mathrm{TH}_{\mathrm{max}} = \frac{n}{T_0}$$

Law (CONWIP with Flexible Labor): *In a CONWIP line with n identical workers and w jobs, where $w \geq n$, any policy that never idles workers when unblocked jobs are available will achieve a throughput level* TH(w) *bounded by*

$$\mathrm{TH}_{\mathrm{CW}}(n) \leq \mathrm{TH}(w) \leq \mathrm{TH}_{\mathrm{CW}}(w)$$

where $\mathrm{TH}_{\mathrm{CW}}(x)$ *represents the throughput of a CONWIP line with all machines staffed by workers and x jobs in the system.*

Law (Variability): *Increasing variability always degrades the performance of a production system.*

Corollary (Variability Placement): *In a line where releases are independent of completions, variability early in a routing increases cycle time more than equivalent variability later in the routing.*

Law (Variability Buffering): *Variability in a production system* will *be buffered by some combination of*

1. *Inventory*
2. *Capacity*
3. *Time*

Corollary (Buffer Flexibility): *Flexibility reduces the amount of variability buffering required in a production system.*

Law (Conservation of Material): *In a stable system, over the long run, the rate out of a system will equal the rate in, less any yield loss, plus any parts production within the system.*

Law (Capacity): *In steady state, all plants* will *release work at an average rate that is strictly less than the average capacity.*

Law (Utilization): *If a station increases utilization without making any other changes, average WIP and cycle time will increase in a highly nonlinear fashion.*

Law (Process Batching): *In stations with batch operations or with significant changeover times:*

1. *The minimum process batch size that yields a stable system may be greater than one.*
2. *As process batch size becomes large, cycle time grows proportionally with batch size.*
3. *Cycle time at the station will be minimized for some process batch size, which may be greater than one.*

Law (Move Batching): *Cycle times over a segment of a routing are roughly proportional to the transfer batch sizes used over that segment, provided there is no waiting for the conveyance device.*

Law (Assembly Operations): *The performance of an assembly station is degraded by increasing any of the following:*

1. *Number of components being assembled.*
2. *Variability of component arrivals.*
3. *Lack of coordination between component arrivals.*

Definition (Station Cycle Time): *The average cycle time at a station is made up of the following components:*

$$\begin{aligned}\text{Cycle time} = {} & \text{move time} + \text{queue time} + \text{setup time} + \text{process time}\\ & + \text{wait-to-batch time} + \text{wait-in-batch time}\\ & + \text{wait-to-match time}\end{aligned}$$

Definition (Line Cycle Time): *The average cycle time in a line is equal to the sum of the cycle times at the individual stations, less any time that overlaps two or more stations.*

Law (Rework): *For a given throughput level, rework increases both the mean and standard deviation of the cycle time of a process.*

Law (Lead Time): *The manufacturing lead time for a routing that yields a given service level is an increasing function of both the mean and standard deviation of the cycle time of the routing.*

Law (CONWIP Efficiency): *For a given level of throughput, a push system will have more WIP on average than an equivalent CONWIP system.*

Law (CONWIP Robustness): *A CONWIP system is more robust to errors in WIP level than a pure push system is to errors in release rate.*

Law (Self-Interest): *People, not organizations, are self-optimizing.*

Law (Individuality): *People are different.*

Law (Advocacy): *For almost any program, there exists a champion who can make it work—at least for a while.*

Law (Burnout): *People get burned out.*

Law (Responsibility): *Responsibility without commensurate authority is demoralizing and counterproductive.*

19.4 A Factory Physics Parable

In this book we have introduced a host of widely varied concepts in order to develop the perspective, intuition, and tools for designing and improving manufacturing systems. To illustrate how many of these factory physics pieces might fit together in a systems analysis project to improve a specific system, we now consider a case study. The scenario is actually a composite of many different companies. Much of the data come from an excellent case by Bourland (1992). However, any lack of literary merit is entirely the responsibility of the authors.

19.4.1 Hitting the Trail

It was 6:20 on a Friday afternoon when Carol snapped her briefcase shut and stood up to go. Her one thought was, *Time to hit the trail!* She had been promised a week's vacation when she joined Texas Tool and Die as manager of manufacturing engineering four months ago. But every time she made plans, a plant crisis forced her to postpone. *Not this time. I've been wanting to go riding in west Texas for years.*

Before she could reach the door, the phone rang. *Not again!* She knew she shouldn't answer it, but her travel agent had said he might call with some last-minute schedule changes. So, gingerly, she picked up the phone.

"Carol Moura."

"Carol. Claude. Good thing you're still here. Milling is out of control again, and Bill wants us in his office *now.* I'll come by."

Carol clapped the phone into the receiver hard. *This will never end!* Not since her freshman year as an engineering student at Michigan State, far from her tight-knit family in Connecticut, had she felt so alone and depressed.

On the way to Bill's office, Claude Chadwick, a production manager, chattered on about the current situation, making sure to stress how critical Carol was to a solution. *Sure. All he wants is for someone to do his work so he can get out this weekend. Him and his marketing MBA. He doesn't care about the plant. It's just a stepping stone to bigger and better things. "Doing my time," he says. As if the plant is a prison.*

Carol's jaw tightened as she spied the sign on the office suite—William Whyskrak, Vice President of Manufacturing. *Bill Whyskrak! "Wiss-krek" he pronounces it. He's forever finding ways to make me look bad. Like that time in printing. First he tells me my cart-sharing idea for reducing cycle times is the stupidest thing he ever heard. Then he gives me a royal chewing out for going ahead with it. But when it worked, he takes all the credit. Worse, he tells Mr. Walker now he'd been trying to get me to do it for weeks and that I had been dragging my feet. Mr. Walker told him to "keep up the good* *work," but only smiled at me. What did that mean? Well, I was looking for a job when I found this one."*

In his office, this time Carol doesn't even give Bill time to explain the latest crisis.

"Bill, I've postponed my vacation three times now. I deserve this time off. If I don't go now, I never will. See you in a week."

That wasn't so hard. On her way to the airport she began to forget the plant. It was early May, the flowers were gorgeous, the weather clear and cool. She let herself relax and started to enjoy the drive. *A week with nothing but my horse, sleeping bag, slicker, and hat to think about. My only problems will be food and water, and there's plenty of that on the wagon. It's going to be a good week.*

Carol spent the first three days on the trail trying not to think about the plant, and mostly succeeding. But on the morning of the fourth day, it forced its way into her consciousness. *What have I really accomplished in four months? A few small things and a lot of crisis management. But I haven't turned things around by a long shot. Bill has no faith in me. Maybe Mr. Walker doesn't either—I can never tell with him. Maybe I won't have a job when I get back. I was looking* hard *for a job when I found this one.*

Bob McAlister, the trail boss, broke her reverie by pulling up to ride alongside her. "Good thing that horse knows where to go."

"What do you mean?" So far, she had had little to do with Bob. He was usually busy making sure everyone's gear was right and had been quiet the rest of the time. Almost all he had said to her was, "Mornin' Ma'am." Even when he checked her saddle girth, all he did was pat the back end of her horse and tip his hat. Bob really seemed to fit the image of the silent cowboy.

"What I mean is that you're not *here.* You're back *there.* If you're going to spend good money to get away from there, why do you want to bring it here?"

"You're pretty smart," Carol admitted.

"You got to have a PhD in psychology to be a trail boss—state law, you know." Bob was the kind of Texan who liked to make outrageous statements with a straight face and see how long it took the non-Texans to catch on. "Trail ridin' takes brains. Your horse ain't gonna tell you he's goin' lame, and that mama cow over there ain't gonna e-mail you she's runnin' dry. It's clear that somethin's botherin' you. Why, you're twitchin' like a long-tailed cat in a room full of rockin' chairs."

Carol laughed. "You're right. I've been wondering if I'll have a job to go back to."

"Maybe I can help. I know you're some kind of big engineer at a plant. I'm no engineer, but you never know, comin' at it from a different angle; I might just see somethin'. Anyway, we got a long way to ride today, and we might as well talk a spell."

"All right, but I'm warning you, it's technical. We make parts and assemblies for aircraft. I'm responsible for making hubs. We get orders..."

Carol talked for 10 minutes before Bob interrupted, "I don't want to know all that. I'm a simple cowboy—just give me the basics. You're tryin' to take one piece of metal and turn it into a different piece, right?"

"Yes, but there are a lot of different pieces..."

"And after you do it, you want to sell the right number of the right piece of metal to the right customer, right?"

"Of course, but there are all kinds of..."

"And you need to do all this with the equipment you got in your plant right now, right?"

"Yes, but..."

"And you want to do it without keepin' your customers waitin' or havin' a lot of extra stock layin' around, right?"

"Yes, but it's a complicated plant. The issues are just not that simple!"

"Who said they were? But I know one thing."

"What's that?"

"Details may not be simple, but *principles* are!" Bob pulled out his canteen, took a drink and offered it to Carol.

Carol took a drink, wiped her mouth, and asked, "OK, what *are* the principles? I've taken every short course there is and have come to the conclusion that for every expert telling me to do one thing, there's another expert telling me to do something else."

"Well, I don't really know."

Carol rolled her eyes. "Great! Maybe I can get a job shoeing horses."

"Wouldn't recommend it. Too hard on your back. What I do know is that there *are* principles and the important ones ain't that hard. You know, like an apple fallin' from a tree. Sometimes the principle is just hidden. You can't see the forest for the trees—that is, if you got trees. Out here I guess it's the hill for the rocks." Bob surveyed the landscape and continued.

"Anyway, a couple years ago, the Extension Service sent out this young expert to make the local feed co-op more efficient." Bob nearly spat out the word *expert*. "By the time he was finished, the place was a mess. I was so mad, I stood up in a meetin' and said a ol' cowpoke like me could've done a better job. Durned if they didn't vote me president that year. Well, I had to do somethin' then. So, I went in, called a meetin' and asked a single question, just one: What in the world is it we're tryin' to do here?

"You should've seen the looks I got. They thought I was dumber than dirt. But when folks started answerin' the question, the place really heated up. We got somethin' like 20 different answers and almost a fight or two. But folks got the picture. Nobody had any idea what we were trying to do. So we sat down, agreed on some goals, and figured out ways to make 'em happen. Actually, it was pretty simple once we got started."

"But what were the principles?" Carol asked. But Bob wasn't looking at her. He was staring at one of the horses near the front of the line.

"Pardon, Ma'am, but it looks like we got a runaway. Talk to you later." Bob spurred his horse and took off after a galloping mare carrying a frightened boy.

Bob stopped the horse and returned the boy to his mother in short order. But his horse had lost a shoe. It stumbled on the way back to the group and threw Bob to the ground. His knee hit a rock and knocked a pin loose from an old rodeo injury. Jedidiah the cook took him to the first ranch house they came to and he was hurried to the hospital. The damage turned out not to be serious, but Bob wouldn't ride again for a month.

After the excitement had died down, Carol began to think about "principles." *If only my problems were that simple. But then, I don't think the co-op problem was all that simple, no matter what Bob says. After all, the "expert" wasn't able to solve it. Maybe most people's problems are just as hard as mine. Maybe everyone has to look for principles of some kind. Like the apple falling from a tree. That's physics. But I have a factory to manage...Wait a minute, what about that factory physics I learned about in B-school? Didn't that have principles that are supposed to be relevant to factories?*

For the rest of the trip, Carol continued to muse about using principles to figure out what was wrong with the plant and fix it. She soon realized she would need help. *Jane Snyder—she was just promoted to manager of marketing—she seems sharp. And Ed Burleson, the manufacturing engineer who came in with me, is a computer whiz. Both strike me as go-getters. What principles do they use? Maybe I can get them together and we can develop a plan. Of course, we can't spend much money. Bill would never go for that. But we could do pretty much whatever we want on the plant floor. No one really pays attention to that—until the end of the quarter—or when customers are screaming.*

I hear they're going to sell the plant. But if we can make the operation run better, we might just keep our jobs.

19.4.2 The Challenge

Texas Tool and Die, which was founded in the 1950s, makes components for the aircraft industry at a single plant near Fort Worth, Texas. Two years prior to Carol's arrival, TTD had been bought out by an investment group that hoped to improve operations and sell it for a profit. An immediate reorganization brought in Bill Whyskrak, a polished speaker with management experience in several industries, and his assistant Claude Chadwick. But despite the changes and a major influx of capital, profits had steadily declined in the face of increasingly stiff competition from firms with lower prices and better customer responsiveness.

The managing owner was a man named Sam Walker, who had started his career as a design engineer and had worked his way into management. Sam was convinced that they had to find ways to increase throughput (to lower unit costs so they would allow more competitive pricing) and to reduce cycle times (so they could offer competitive customer deliveries). He directed Bill to bring in more manufacturing talent—which led to the hiring of Carol Moura, a manufacturing engineering manager with 10 years of experience and an MBA in operations, and Ed Burleson, a manufacturing engineer with a BS in industrial engineering. Two months after Carol and Ed came on board, things had gotten so bad that some of the investors were at the point of wanting to sell the company, take their losses, and move on. Sam convinced the other owners to give the throughput enhancement and cycle time reduction efforts one more chance. The other owners agreed to six more months of operations, with the stipulation that no large capital expenditures be made.

19.4.3 The Lay of the Land

Historically, company policy had been to collect customer orders during the week and group them into jobs every Friday. In its product catalog, TTD promised delivery four weeks after the close of business on Friday. Unfortunately, the competition was offering three-week lead times and had been steadily reducing these each year. Worse, TTD had not been able to achieve even the four-week target with regularity. Average cycle time for some parts was well over eight weeks.

Although average demand was still high, it was variable, to the point that there were times when there was almost no demand for the week. Figure 19.1 shows the aggregate demand for the previous year. Table 19.1 gives projected demand for the next year for the four largest-selling products, which accounted for 90 percent of total demand, along with the lot size for each product. Demand for other products was met by production from a job shop separate from the part of the plant that produced hubs 1 through 4.

Several months before Carol and Ed had arrived, Bill and Claude had organized the main processes for producing hubs 1 to 4 into a cellular layout in an attempt to reduce cycle times by eliminating unnecessary material handling. The anticipated reduction had yet to materialize. The cell consisted of three benches (which served as preparation stations), four vertical lathes (VTL), one deburring station, four inspection stations, two mills, two drills, and one rework station. All machines were subject to occasional breakdown. Table 19.2 gives data gathered on mean times to failure and mean times to repair.

There were 14 workers in the cell, with three prep workers assigned to the benches, three repair operators assigned to the deburr and rework stations, three inspectors

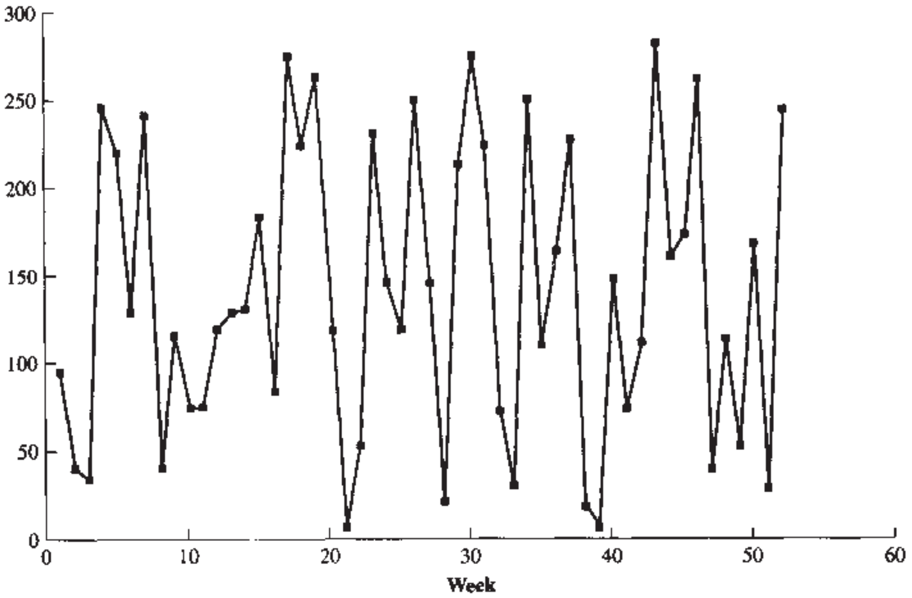


Figure 19.1
Total demand for previous year

Table 19.1 Average Demand and Lot Sizes

Part	Average Demand	Lot Size
Hub 1	2,100	40
Hub 2	1,700	30
Hub 3	2,000	44
Hub 4	1,500	30

assigned to the inspection stations, and five machinists assigned to the lathes, drills, and mills. Figure 19.2 shows the layout of the facility, along with the labor assignments. Due to breaks—scheduled and unscheduled—workers were generally considered available only 90 percent of the time.

The sequence of operations (routing) for hub 1 is shown in Figure 19.3. Run times, setup times, and labor times are given in Table 19.3. Because many of the operations were automated, the labor time for some operations was less than machine time, so it was possible for an operator to monitor multiple machines. The routings and process times for the other products were similar to those for hub 1.[1]

As Figure 19.3 shows, an average of 15 percent of the hub 1 parts were found to be defective at the inspection station. An average of two-thirds of these were sent to rework; the others were scrapped. In rework, an average of 20 percent were reworked without success and were eventually scrapped. The remaining 80 percent were reworked and sent back to inspect, where they might or might not be certified as good parts.

[1] The details of all the parts are not central to our story. The interested reader is referred to Bourland (1992) for other details of the case.

Table 19.2 Equipment Data

Equipment Group	Number in Group	Reliability		Labor Group Assigned
		MTTF (hour)	MTTR (hour)	
Bench	3	160	8	Prep
VTL	4	160	16	Machinist
Deburr	1	80	8	Repair
Inspect	4	40	8	Inspector
Repair	1	160	8	Repair
Mill	2	80	4	Machinist
Drill	2	160	4	Machinist

Figure 19.2

Cell layout

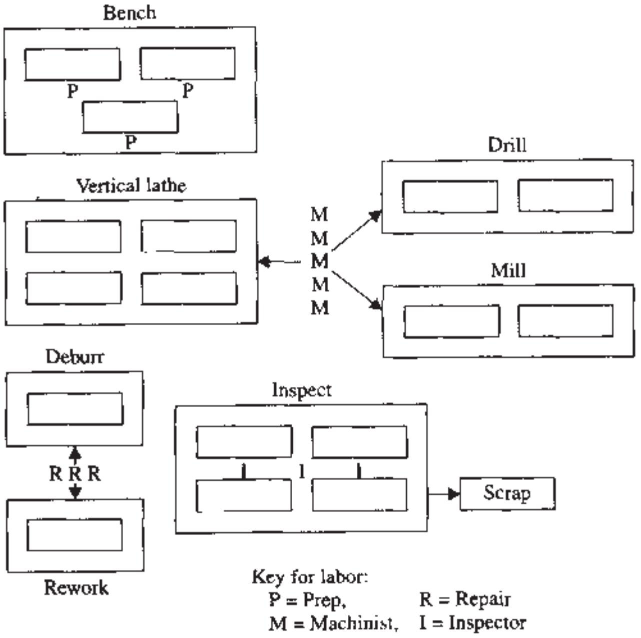
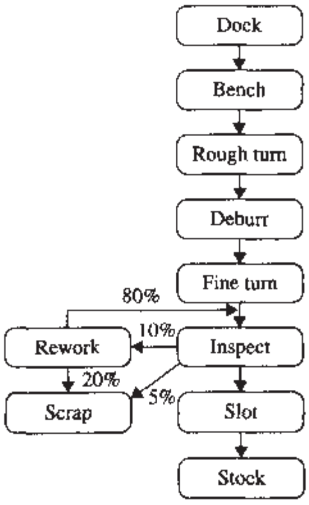


Figure 19.3

Operations and routings



Each hub was composed of four to six mountings and a single sleeve. Each mounting was composed of two brackets and two bolts. The brackets, bolts, and sleeves were all purchased from outside suppliers. Since these parts were common to many assemblies, TTD tended to keep ample stocks of them. Table 19.4 gives the process times for the unpacking and inspection of the purchased parts. The assembly of the mounts, sleeves, and hubs took place in the assembly area, which seemed to have sufficient capacity and rarely failed to keep up with the cell.

Table 19.3 Operation Assignments and Process Times for Hub 1

		Time at Equipment		Labor Times	
Operation	**Equipment**	**Setup Time (minute)**	**Run Time (minute/piece)**	**Setup Time (minute)**	**Run Time (minute/piece)**
Bench	Bench	0	10	0	10
Rough turn	VTL	180	17	180	15
Deburr	Deburr	0	10	0	10
Finish turn	VTL	120	26	120	20
Inspect	Inspect	7	12	7	7
Rework	Rework	90	32	90	32
Slot	Mill	60	60	60	40

Table 19.4 Operation Assignments and Process Times for Purchased Parts

		Time at Equipment		Labor Times	
Operation	**Equipment**	**Setup Time (minute)**	**Run Time (minute/piece)**	**Setup Time (minute)**	**Run Time (minute/piece)**
Mounting					
Unpack	Bench	12	2	12	2
Inspect	Inspect	0	3	0	3
Bracket.					
Unpack	Bench	12	0	12	0
Inspect	Inspect	10	0	4	0
Bolt					
Unpack	Bench	12	0	12	0
Inspect	Inspect	12	0	4	0
Sleeve					
Unpack	Bench	12	3	12	3
Inspect	Inspect	0	3	0	3

19.4.4 Teamwork to the Rescue

Carol returned from her vacation rested but anxious. There were seven progressively shrill calls from Bill Whyskrak on her voice mail. *Big surprise.* Before returning them, she called Jane Snyder and Ed Burleson—who both agreed that the plant was in big trouble—and asked them to meet her after work at the local watering hole. They agreed. Then she called Bill and endured another haranguing.

No sooner had she hung up than Claude slithered into her office with his version of the past week's disasters and bitter complaints about having to work all weekend. *About time!* When he had gone *(Finally!),* Carol moved the pile of unanswered mail to the side of her desk *(It'll keep one more day),* got out her old *Factory Physics* text *(Dusty but it* *still looks almost new),* and began looking for "principles." When it was time to go to the bar, she was ready.

Principles. "What in the world is it that we're trying do do?" Carol asked as she, Jane, and Ed waited for the beer and nachos to arrive. After some discussion of basic concerns like "keep our jobs," the three agreed that two fundamental problems were driving costs up and revenues down: insufficient throughput and excessive cycle times. If they could make a significant difference in these, they believed TTD could be made profitable.

Carol had anticipated this and was armed with some principles from *Factory Physics.* She began by pointing out that Little's Law shows that throughput and cycle times are related:

Law (Little's Law):

$$\text{WIP} = \text{TH} \times \text{CT}$$

"Cool!" Ed observed. "If we can get throughput up to capacity and keep it there, then reducing WIP will reduce cycle time."

"Exactly!" Carol knew there was a reason she had asked Ed along. "Except that we have to be careful about aiming for capacity." She displayed her next factory physics law.

Law (Capacity): *In steady state, all plants* will *release work at an average rate that is strictly less than the average capacity.*

"Okay. That's what I meant, actually. Everyone knows that machines can't run all the time."

"Oh yeah?" Jane raised her eyebrows. "How many times have you heard Bill screaming for 100 percent utilization of the lathes? But if we're going to talk about principles, let's leave Bill out of it." Ignoring Ed's groan, Jane went on. "Carol, I'm wondering about that Little's Law. It looks like we can get the same throughput with small WIP and small cycle times or big WIP and big cycle times. It's pretty clear which category we fall into, but what's the difference?"

"I couldn't have set it up better myself." Carol smiled and presented her next law.

Law (Variability): *Increasing variability always degrades performance of a production system.*

"And I found one more that follows up on the variability theme."

Law (Variability Buffering): *Variability in a production system* will *be buffered by some combination of*

1. *Inventory*
2. *Capacity*
3. *Time*

"The book also refers to this as the pay-me-now-or-pay-me-later law," she said.

"Nice name," grinned Ed. "But what's it mean?"

"It means we have either too much variability or too much WIP. But if we keep WIP too low, we lose on throughput and so we have a capacity buffer," Carol explained.

"How could we be keeping WIP too low? I thought we had too much WIP."

"Whenever we turn off releases because WIP has gotten out of hand, we lose throughput."

"You mean like the week you were gone."

"Uh huh. But before we can even talk about a reasonable target throughput, we need to know what our capacity is."

"How do we do that?"

"You guys up for a walk? Let's go back to the plant," Carol suggested, as she picked up the check.

The scene at the manufacturing cell was all too familiar. The trio found WIP piled high in front of the bench operation, vertical lathes, and the milling machines. Things were so bad that the prep workers had just returned a load of materials to the storeroom to relieve the congestion. The machinists were complaining that they were being overworked again as the repair operators were "just sitting around." When questioned, an idle repair operator explained that his load was sporadic; he couldn't help it if he sometimes ran out of work to do.

"We've got our work cut out for us," said Ed as they walked out to the parking lot.

"But where do we start?" asked Jane.

Carol reached her car first and unlocked the door. "I suggest we listen to the machinists. Maybe they *are* overworked. I'm going to run some numbers. Let's talk about it tomorrow, okay Ed? Night, Jane."

"Night."

Capacity Analysis. The next morning, Carol set up a spreadsheet and did a quick estimate of the utilization levels of the machinists and repair operators. She did this by calculating the total load generated by production needed to meet demand, including setups, at the current lot sizes. This showed that the average workload of the machinists was indeed higher than that of the repair operators. Ed determined that one repair operator could be moved into the machinist pool without compromising the ability of the repair operators to do their work. Fortunately, one of the operators had worked as a machinist, was bored with his repair job, and welcomed the move. Since no one could come up with a reason not to, Carol talked the foreman into making the switch that afternoon.

Cycle Time Analsyis. What to do next was not so obvious. Carol's simple spreadsheet did not suggest any more easy labor reassignments, and no one could offer a clear idea of how variability was affecting the system. Almost for lack of anything else to do, Ed volunteered to develop a simulation of the facility. After a week of coding, debugging, and preliminary runs, he had a basic working model. He was pleased to be able to show Carol and Jane that his simulation predicted extremely long (indeed unstable) cycle times in the cell when staffed by three repair operators and five machinists. However, if one repair operator were reassigned, so that there would be two repair operators and six machinists, the simulated cycle times dropped to between four and seven weeks, with hub 1 having the longest.

"It looks like we did the right thing," he concluded with a grin. "Cycle times should be coming down soon."

And for a while the system really did seem to be improving. Two weeks after reclassifying the repair operator as a machinist, throughput was up noticeably. But cycle times were still well above the levels predicted by the simulation. The team was puzzled at the discrepancy and rechecked the process times on the machines. The times used in the simulation were found to be, if anything, longer than those observed in the actual system.

"It's not the rate data." Ed looked up from his keyboard. "What else could be making the cycle times so much longer than the model says they should be? Do we have any other data we could check?"

"Not many," Carol admitted. "But we do have these WIP sheets. What does the simulation say about WIP?"

"I don't know. I'll run it again and generate WIP-versus-time charts for the different equipment groups."

"Good. I'll make up the same charts from these sheets. Let's meet for coffee around four. I'll call Jane."

Four o'clock found the team members hunched over a cafeteria table, studying the two charts. They did not look anything alike. The simulation model predicted fairly modest increases and decreases in WIP, while the actual WIP charts showed huge "bubbles" of WIP that drifted through the plant.

"What's causing that?" Jane asked.

"Queueing," Carol answered.

"What's that equation for queue time again?" Jane reached for the no-longer-dusty copy of *Factory Physics*.

"Whoa!" Ed feigned falling out of his chair. "A marketing person asking for an equation!"

"Give me a break! Marketing *is* quantitative, you know. Here it is."

$$\mathrm{CT}_q = \underbrace{\frac{c_a^2 + c_e^2}{2}}_{\text{Variability}} \times \overbrace{\frac{u}{1-u}}^{\text{Utilization}} \times \underbrace{t_e}_{\text{Process time}}$$

Jane studied the formula carefully and mused, "Hmmm. Since our process times are conservative, utilization must also be conservative, since the throughput is right."

"Wow! I guess you marketing types do know your way around an equation," said Carol, obviously impressed.

"So it must be in the variability numbers," Ed added swiftly, not wanting to be outdone in the technical analysis department.

"Which one?" Jane asked.

"Well, the *c*-sub-*e* number could be big, but not that big. And I don't see how the *c*-sub-*a* number can get very big either," Carol said with a puzzled look.

"What are *c*-sub-*e* and *c*-sub-*a*?" asked Jane.

"The *c*-sub-*e* is a measure of how variable the machine process times are, while the *c*-sub-*a* measures the variability of arrivals," Ed explained, a little relieved to have an opportunity to display his knowledge.

"What does it mean for arrivals to be variable?"

"If they don't come in one at a time, regularly, like clockwork, then they're variable."

"Well, of course they don't come in like that. We release jobs in week-long batches. It's part of our marketing strategy," Jane explained.

"Hello!" Ed grinned. "Maybe you better tell us more about that strategy."

"We publish a lead time to our customers. Any order we get during a given week will be delivered four weeks later. The close-out day is Friday. Orders are batched over the weekend and then sent to the floor on Monday. We've been doing it for years. Efficiency considerations. you know."

"Well, it might make things more efficient, but I'll bet it's driving the heck out of cycle time. No wonder we see all these WIP bubbles." Carol said and turned to Ed. "What *c*-sub-*a* do we have in the model?"

"For lack of a better number, we used one, the usual exponential assumption." Ed snuck a glance over at Jane to see if this technical talk was making her nervous. It wasn't.

"Probably way too low. My guess would be more like 10."

"It might even be worse," Jane added. "There's a lot of variability in our demand as well. Take a look at this."

The chart (Figure 19.1) showed that total weekly demand for the past 12 months averaged 146 pieces, but ranged between 6 and 284. Thus, while the capacity of the plant was around 160 parts per week, it was faced with a "feast or famine" situation. Clearly, this meant that in some weeks the plant was starved for work, while in others it was completely swamped.

Ed stood up. "I've got to change the way I model demand. I'll talk to you tomorrow."

Carol accompanied Jane back to her office. "Jane, what would happen if, instead of publishing a fixed lead time, we quoted delivery dates to our customers. And what if those dates were closer in than four weeks?"

"Well, getting lead times below four weeks would be great. The competition is killing us on that. And I guess most of our customers would probably like a quotation better—provided we deliver on time. But some customers have their MRP system loaded with our lead time. Could we have a fixed lead time for them?"

"I think so, at least most of the time. But when we're really busy, we may not be able to meet the fixed lead times."

"Actually, now that I think of it, that might not be so bad. Usually, when we're swamped, so are our competitors."

"Good point. The main thing, though, is that we'll be able to quote shorter lead times on average."

"Our customers will like that. What do we need to do?"

"It's called *due date quoting*, and we can do it for each of our product lines. This gives some details." Carol handed Jane the *Factory Physics* book. "See the chapter on scheduling."

"All right, I'll get on it."

The next morning, Ed was in Carol's office early.

"Got it! I changed the arrival processes, and the simulation matches on cycle times pretty well. Now what?"

"Now we get rid of those WIP bubbles."

"How?"

"Well, I think a pull system will smooth the workload. I'll work on that. You see if you can find ways to reduce process variability. Okay?"

"Sounds like a plan."

During the next month, Carol set up a CONWIP system in the cell. The mechanics were simple, basically consisting of nothing more than laminated cards to limit WIP and the standard work list to sequence releases. More challenging was breaking the tradition of bulk releases. Carol carefully involved the operators in the implementation process, and even shut down the cell for a two-hour "all hands" orientation meeting. (She thought Bill was going to burst a vein over that!) To the operators, CONWIP seemed almost obvious; after all, why release work into the cell until there is capacity to work on it? A couple of people in production control, who were responsible for running the MRP system that scheduled the bulk releases, initially raised some objections about having their schedules overridden by the CONWIP system. But Jane helped Carol win them over, by stressing the marketing value of shorter cycle times.

Meanwhile, Ed searched his simulation and the cell for large sources of variability in effective process times. At first, the process times seemed extremely regular, since processes were largely automated. Then he realized that he needed to consider the effect of downtimes that averaged from 4 to 16 hours on the various machines. Ed performed a Pareto analysis of previous failures and found that most of the maintenance calls were the result of a small set of problems. He and the maintenance superintendent developed efficient procedures for handling the most common problems and then documented them. Where appropriate, they also installed field-ready replacement kits. The result was that mean time to repair on all machines dropped to less than four hours. Although they would not have data to document it for months, the beneficial effects on the line were felt almost immediately.

After the blowup about Carol's CONWIP meeting, Bill mysteriously emerged as a convert to JIT. He gave Carol and Claude a popular JIT book and ordered Carol to install a kanban system in the cell and Claude to implement JIT deliveries of raw material. Carol ignored the book, but was careful to refer to her CONWIP system as a kanban system whenever she spoke to Bill. Luckily for her, Bill didn't have time to pay too much attention to what she was doing because of problems with Claude's policies.

With Bill's blessing, Claude changed from purchasing commonly used pieces of bar stock in one-month supplies to having daily deliveries from a local vendor. Raw material inventory dropped by 80 percent, but delivery charges went up dramatically as well. Bill stepped in and threatened to cancel the contract because of the higher delivery cost. The offended vendor responded by canceling the contract himself. The production schedule was badly scrambled, and production came to a virtual halt for almost two days before Sam Walker smoothed things over with the vendor and reestablished the supply.

Also at Bill's instigation, Claude began a plantwide setup reduction program that made use of single minute exchange of die (SMED) techniques Ed had developed previously for a specific machine. Because these techniques did not apply universally and because effort was spread over so many processes, Claude got off to a slow start. By mid-July, after almost two months' work, he had achieved significant setup reductions only in the labeling area. However, about the time Claude's program was beginning to stall, Ed became convinced from his ongoing simulation study that setup reduction was important on the VT lathe, drilling, and milling. He took over (unofficial) leadership of this part of the program, and by the end of August they had reduced the setup times of the VT lathe, drilling, and milling by 50 percent. With these and the other changes they had made, Ed's model predicted cycle times of 9 to 22 days, compared with the original 5 to 9 weeks.

At the next team meeting, Carol copied the basic cycle time equation from the increasingly ragged copy of *Factory Physics* to the board:

Definition (Station Cycle Time): *The average cycle time at a station is made up of the following components:*

$$\begin{aligned}\text{Cycle time} = {} & \text{Move time} + \text{queue time} + \text{setup time} + \text{process time}\\ & + \text{wait-to-batch time} + \text{wait-in-batch time} + \text{wait-to-match time}\end{aligned}$$

"The way I see it, CONWIP and due date quoting have brought queue times down by something like 80 percent. Process times and move times were never big. Wait-to-match time doesn't apply in the cell. So, the only remaining area to be addressed is wait-for-batch time." Carol sat down. "Ed, what move batch sizes are we using in the model?"

"The ones they use in the plant. They were computed using the square root formula. I think. Why?"

"So the batch sizes are the same for both move batches and process batches?"

"What do you mean by *move batch* and *process batch?*" Jane asked. "I've never heard anyone here use those terms."

"That could be our problem." Carol answered. "The process batch is how many parts we run between setups. The move batch is how many we move at once to the next operation. They don't have to be the same."

"Why didn't I think of that!" Ed began sliding his chair back. "Let me see what happens in the model if we leave our process batch sizes alone but make all the move batches in the cell equal to one."

"Wait. Let me get this straight," Jane jumped in before Ed could escape. "You mean, like for hub 1, we process 40 units before changing over to another hub but move them one at a time as soon as they're done?"

"Exactly!"

Carol was confident that she knew what Ed's simulation would show. Smaller move batches would result in shorter cycle times. But while she was waiting for him to estimate the size of the reduction, Carol began thinking about the process batch sizes. *Since we reduced setup times, we should be able to reduce batch sizes as well. But how much? That silly EOQ formula won't help because we have no idea what setup cost should be. Besides, the interaction between the batch sizes of the various hubs is probably complex. Wasn't there something in the scheduling chapter about optimal batch sizing to minimize cycle times?*

She picked up the phone to call Ed, but he walked in before she had a chance to dial.

"Good news! The cycle times should drop another 30 percent by simply making the move sizes equal to one. But I think we could do even better if we adjust the process batch sizes, so I started reading in Chapter 15 about..."

"Optimal process batch sizes! You're reading my mind. I was just calling you to suggest we fiddle with process batch sizes."

Carol and Ed spent a few hours building an optimal batch-sizing model. Using it along with some trial-and-error, they settled on the set of batch sizes shown in Table 19.5. The next morning Ed met with the shop superintendent, who readily agreed to the changes in process and move batch size. Congestion in the cell steadily declined. By the end of September, cycle times had fallen to between four and seven days.

19.4.5 How the Plant Was Won

October was judgment time. Sam Walker gave Bill responsibility for organizing an overview of the improvement program at a meeting of the owners. Bill told Carol and Claude that he'd handle the presentation himself. Carol made up some slides anyway, just in case. Claude did not.

TABLE 19.5 Recommended Batch Sizes and Resulting Cycle Times

Part	Recommended Batch Size	Predicted Cycle Time
Hub 1	10	6.7
Hub 2	15	3.4
Hub 3	20	5.6
Hub 4	15	3.7

Sam began the meeting with a brief overview of how much output had increased, cycle times had decreased, and customer relations had improved. He concluded with, "And now I'm going to ask Bill to tell us just what was done to make this good news possible. Bill?"

Bill was dressed to the nines and had slick color slides. A couple of owners even laughed at his introductory jokes. *He's going to pull this off! All the work we did, and we won't get a shred of credit.* Carol sighed as Bill moved into the core of his presentation.

"The key to our cycle time reduction program was recognizing what cycle time is." Bill put up his main slide, which showed:

$$\text{Cycle time} = \text{Value-added time} + \text{non-value-added time}$$

"Things like setup time, move time, unnecessary meeting time," Bill emphasized the last item with a glance at Carol, "are all waste. Or, as they say in Japan, *muda*. Eliminate *muda* and you'll reduce cycle times." Bill flipped up the next slide. "One of our most successful efforts was reducing setups through the use of SMED techniques. Take labeling for instance..."

"Wait a minute, Bill," Sam interrupted. "Why do we want to reduce setup times in labeling? We've got plenty of capacity there, and I've never seen much WIP in that area. What's the point?"

"Well, as I said, setups represent non-value-added time. They should be eliminated."

"Is that what you were doing last winter in printing? I recall that once you got Carol going, you eliminated a cart at each table and had the operators share a single cart. Seems to me like you added quite a bit of walking around. Isn't that non-value-added?"

Got Carol going! Carol's heart sank. *He thinks I'm in the way!*

"Well, er, it depends. In this case...," Bill's polished demeanor faltered just a bit. "Claude, didn't you want to say something about our lean manufacturing program to Mr. Walker?"

Carol watched the panic rise in Claude's face. *Well, at least I'm not the only one Bill makes look bad.* But Claude covered neatly.

"Well, I think it's pretty clear that the proof's in the pudding. As you can all see, Bill's program has really turned things around." Claude turned from Bill to Sam. "Regardless of what you call it. After all, we're here to run the plant, not name things."

Some of the owners nodded in agreement. Sam was noncommittal and quickly looked back to Bill. "Wasn't there more to the program than setup reduction?"

"Yes. You'll recall that we also implemented just-in-time deliveries."

"I remember," muttered Sam under his breath.

"And we installed a simple kanban system in the cell that increases efficiency by pulling parts between machines and..."

"Excuse me Bill," Sam interrupted again. "I've been down to the cell and I believe I've heard the operators referring to the new system as CONWIP, not kanban. Why is that?"

"Oh! Well,..., it's basically the same thing. Actually, Carol helped me quite a bit with that part, so maybe we should ask her."

Carol swallowed hard and walked up to the projector.

"*CONWIP* stands for constant work in process and is *not* quite the same thing as what most people mean by kanban..." Carol gathered steam as she spoke. She rolled through the importance of variability, the effects of batching, and even put up a few factory physics graphs. She showed plots of the progressively shorter cycle times predicted by the simulation model as improvements were incorporated. Her speech grew more rapid, her gestures more animated. Before she knew it, she had spoken for 20 minutes without a single interruption. She stopped and looked up anxiously for questions. The room was silent.

"Thank you, Ms. Moura." Sam had a sly smile on his face.

What can that mean? I must have talked too much, and I shouldn't have contradicted Bill. Now I've done it!

"Thank you all. This is a fine piece of work. Now, if you'll excuse us, I need to wrap up with the owners." Sam motioned them to the door.

As she filed out with Bill and Claude, Carol could hear the owners congratulating Sam. One was shaking his hand, and Sam was smiling broadly.

"I think that went well," said Bill as soon as they were in the hall. "Except for you boring them with your quantoid stuff, Carol. Kanban, CONWIP—nobody cares! But at least we're still in business."

"Yeah." Carol didn't want to join the post mortem with Bill and Claude. "I've got to take care of some things. See you."

Forty-five minutes later, back in her office, Carol was mechanically answering e-mail when the phone rang. It was Sam. They wanted her back in the conference room. Filled with dread, she went.

"Hello, Carol." Sam offered her a seat. "We've been working on a few changes of our own." He flipped on the overhead projector, revealing an organization chart. Carol hastily scanned it for her position. It was unfilled. *Oh no! Well, I did it this time. Now I* am *looking for a job! Me and my big mouth!*

One of the owners said, "Congratulations, Carol!"

Congratulations!? Why that sarcastic... Carol looked back at the screen. In the box labeled VP Manufacturing was her name. Next to it in the position of Manager, Manufacturing Engineering was the name of Edward Burleson. Jane Snyder was listed as VP Marketing for the division.

Sam read the question in her eyes. "We have already discussed matters with Mr. Whyskrak, and he and Mr. Chadwick have decided to leave the company to form their own concern."

Carol sped down the hall in search of Ed and Jane. This called for more than beer and nachos!

19.4.6 Epilogue

Carol was unpacking in her new office. She pulled out the battered copy of *Factory Physics,* with its dog-eared pages and broken spine, and placed it gently on the shelf. *This is about to fall apart. I need a new copy. I sure hope it's still in print.*

When she had emptied and disposed of the boxes, she began sifting through her mail. She spied a piece with a familiar name on it.

Whyskrak & Company
"We add value by eliminating waste."

Sounds good to me! She tossed the flyer into the waste paper basket.

Then she pulled an old card from her organizer and dialed the number. After a pause she said, "Bob? This is Carol Moura from Texas Tool and Die. Remember our discussion about principles?"

19.5 The Future

This book has focused on manufacturing management, within the scope of operations, and using factory physics as the unifying perspective. It is fitting that we close with an assessment of what factory physics is and what we can expect from it in the future.

1. *Factory physics is a start to a science of manufacturing.* We have argued that a science of manufacturing is needed to enable managers to judge which policies will be effective in their system and which will not. In the past 30 years or so, manufacturing has been besieged by one "revolution" after another—MRP, JIT, TQM, TBC (time-based competition), BPR (business process reengineering), SCM (supply chain management), and so on—each of which has undoubtedly contained useful insights. But because each presents only a specific perspective, generally sold in fire-breathing revolutionary rhetoric and justified primarily in terms of anecdotal evidence, the manufacturing manager has no basis on which to choose between them, combine features of different approaches, or develop a unique system adapted to the particular environment. Only a science that describes the critical behavior and interactions in a manufacturing system can provide the over arching understanding needed for this.

Our efforts in this book at the development of a science of manufacturing are far from complete. However, we feel that we have at least framed the problem in the correct context. While we have relied on mathematical formulas, we have *not* sought a "factory mathematics." Our focus has consistently been on the physical behavior of manufacturing systems; mathematics are simply the language for describing this behavior precisely. For example, the basic factory dynamics formulas of Chapter 7 were developed in response to the question, How do WIP, throughput, and cycle time depend on one another? By making various assumptions about the behavior of the plant (e.g., the best case, worst case, and practical worst case), we were able to develop formulas for the curves of throughput versus WIP and cycle time versus WIP. These relationships sharpened our insight into questions like why many plants have excessive WIP levels, why variability reductions can reduce cycle times, and how improvements in a production line can be characterized. However, these formulas are certainly not the final word on the WIP, throughput, and cycle time relationships. In Chapter 12, we returned to these curves and showed that when scrap loss is considered, throughput may eventually decrease in the WIP level—something that our cases in Chapter 7 did not allow.

Because manufacturing systems are complex and diverse, some systems undoubtedly exhibit types of behavior that we have not described in this book. Indeed, as we write this, considerable research is being devoted to describing many different production systems (see Askin and Standridge 1993; Buzacott and Shanthikumar 1993; and Graves, Rinnooy Kan, and Zipkin 1993 for good, up-to-date summaries). Thus, in the next few years, we can expect the range and depth of factory physics to expand significantly. Although advances in manufacturing science will never enable manufacturing management to become merely an analytical exercise, our hope is that it will become more like medicine (i.e., science-based, with a strong human element) and less like fashion (i.e., trendy, without guiding principles).

2. *Factory physics is a pedagogical framework for conveying;*
   a. *Basics*
   b. *Intuition*
   c. *Synthesis*

To give precise descriptions of factory behavior under various conditions, we need appropriate tools (e.g., statistics, queueing theory, reliability). In a factory physics framework, therefore, these become important not just for their own sake, but as building blocks for answering fundamental questions about how plants behave.

We have repeatedly stressed that sound intuition is perhaps the single most important skill of the manufacturing manager, enabling him or her to focus attention on the areas of greatest leverage. By describing the natural tendencies of manufacturing systems, factory physics provides a structure within which to build intuition. The manager who understands factory physics principles and can interpret empirical observations in terms of them will acquire insight into the behavior of a system far more rapidly than a manager without these skills.

We have also stressed that manufacturing systems are complex, multifaceted organizations involving many different processes, people, and machines, and multiple objectives. In such environments, the major opportunities for improvements often lie at the interfaces (e.g., between sales and manufacturing, or between product development and manufacturing). By providing a general description of the manufacturing system, factory physics gives us a means for evaluating the impacts of external changes on plant behavior. As such, it represents a linking mechanism between manufacturing and other business functions.

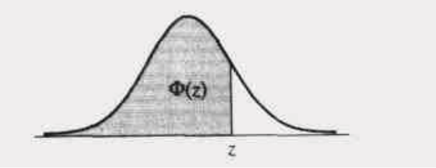
3. *Factory physics is a link between the* process *and* systems *views of manufacturing.* Manufacturing specialists tend to come in two varieties. One group focuses on the specific processes involved in manufacturing, such as robotics, surface finishing, grinding, injection molding. The other group (to which the authors belong) focuses on systems, such as scheduling, inventory control, production planning. Clearly, both sets of concerns are critical to effective operation of a plant. Unfortunately, members of each group are inclined to act as if their view of manufacturing were the only "correct" one. As a result, processes are chosen with little regard for systems impact, and systems are designed with little detailed consideration of processes. Factory physics uses process-oriented descriptors (e.g., mean time to failure, mean time to repair, setup time), condensed into logistics-oriented descriptors (e.g., mean and SCV of effective processing times), to estimate systems-oriented measures (throughput, WIP, cycle time). Thus, it provides a means for interpreting process changes in systems terms.

4. *Factory physics is a collection of tools for quantifying tradeoffs.* As we have seen, increasing capacity, reducing scrap, improving reliability and maintainability, reducing or externalizing setups, upgrading the quality of purchased parts, more frequent moves of smaller batches, and many other policies can have related logistical impacts. By combining the factory physics tools for evaluating these effects with estimates of costs, we can examine the relative attractiveness of each. Moreover, by using the plant-level measures provided by factory physics under different configurations, we can generate cost versus performance curves (e.g., throughput versus cost or cycle time versus cost) and determine strategically desirable targets.

Finally, from an impact standpoint, it is difficult to overstate the importance of factory physics. Roughly one-half of the U.S. economy (jobs, as well as GNP) still depends on manufacturing. Indeed, operational improvements in the manufacturing sector were instrumental in the productivity gains that drove the economic boom of the 1990s. But as competitiveness in the world of manufacturing continues to escalate, the ability to deliver diverse products with high quality, low cost, swift delivery, and reliable service is fast evolving from a recipe for success to a requirement for survival. In the past it was possible to develop effective manufacturing practices by trial and error. In the future there won't be time. Only by sustaining a rapid cycle of continual improvement through the use of principles to quickly develop practices that support strategy will firms be able to keep pace. In the 21st century, mastery of the concepts of factory physics will be as vital a core manufacturing competency as the concepts of mass production were in the 20th century.

TABLE Cumulative Probabilities of the Standard Normal Distribution

Entry is area $\Phi(z)$ under the standard normal curve from $-\infty$ to z.



z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
-3.4	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0003	0.0002
-3.3	0.0005	0.0005	0.0005	0.0004	0.0004	0.0004	0.0004	0.0004	0.0004	0.0003
-3.2	0.0007	0.0007	0.0006	0.0006	0.0006	0.0006	0.0006	0.0005	0.0005	0.0005
-3.1	0.0010	0.0009	0.0009	0.0009	0.0008	0.0008	0.0008	0.0008	0.0007	0.0007
-3.0	0.0013	0.0013	0.0013	0.0012	0.0012	0.0011	0.0011	0.0011	0.0010	0.0010
-2.9	0.0019	0.0018	0.0017	0.0017	0.0016	0.0016	0.0015	0.0015	0.0014	0.0014
-2.8	0.0026	0.0025	0.0024	0.0023	0.0023	0.0022	0.0021	0.0021	0.0020	0.0019
-2.7	0.0035	0.0034	0.0033	0.0032	0.0031	0.0030	0.0029	0.0028	0.0027	0.0026
-2.6	0.0047	0.0045	0.0044	0.0043	0.0041	0.0040	0.0039	0.0038	0.0037	0.0036
-2.5	0.0062	0.0060	0.0059	0.0057	0.0055	0.0054	0.0052	0.0051	0.0049	0.0048
-2.4	0.0082	0.0080	0.0078	0.0075	0.0073	0.0071	0.0069	0.0068	0.0066	0.0064
-2.3	0.0107	0.0104	0.0102	0.0099	0.0096	0.0094	0.0091	0.0089	0.0087	0.0084
-2.2	0.0139	0.0136	0.0132	0.0129	0.0125	0.0122	0.0119	0.0116	0.0113	0.0110
-2.1	0.0179	0.0174	0.0170	0.0166	0.0162	0.0158	0.0154	0.0150	0.0146	0.0143
-2.0	0.0228	0.0222	0.0217	0.0212	0.0207	0.0202	0.0197	0.0192	0.0188	0.0183
-1.9	0.0287	0.0281	0.0274	0.0268	0.0262	0.0256	0.0250	0.0244	0.0239	0.0233
-1.8	0.0359	0.0352	0.0344	0.0336	0.0329	0.0322	0.0314	0.0307	0.0301	0.0294
-1.7	0.0446	0.0436	0.0427	0.0418	0.0409	0.0401	0.0392	0.0384	0.0375	0.0367
-1.6	0.0548	0.0537	0.0526	0.0516	0.0505	0.0495	0.0485	0.0475	0.0465	0.0455
-1.5	0.0668	0.0655	0.0643	0.0630	0.0618	0.0606	0.0594	0.0582	0.0571	0.0559
-1.4	0.0808	0.0793	0.0778	0.0764	0.0749	0.0735	0.0722	0.0708	0.0694	0.0681
-1.3	0.0968	0.0951	0.0934	0.0918	0.0901	0.0885	0.0869	0.0853	0.0838	0.0823
-1.2	0.1151	0.1131	0.1112	0.1093	0.1075	0.1056	0.1038	0.1020	0.1003	0.0985
-1.1	0.1357	0.1335	0.1314	0.1292	0.1271	0.1251	0.1230	0.1210	0.1190	0.1170
-1.0	0.1587	0.1562	0.1539	0.1515	0.1492	0.1469	0.1446	0.1423	0.1401	0.1379
-0.9	0.1841	0.1814	0.1788	0.1762	0.1736	0.1711	0.1685	0.1660	0.1635	0.1611
-0.8	0.2119	0.2090	0.2061	0.2033	0.2005	0.1977	0.1949	0.1922	0.1894	0.1867
-0.7	0.2420	0.2389	0.2358	0.2327	0.2296	0.2266	0.2236	0.2206	0.2177	0.2148
-0.6	0.2743	0.2709	0.2676	0.2643	0.2611	0.2578	0.2546	0.2514	0.2483	0.2451
-0.5	0.3085	0.3050	0.3015	0.2981	0.2946	0.2912	0.2877	0.2843	0.2810	0.2776
-0.4	0.3446	0.3409	0.3372	0.3336	0.3300	0.3264	0.3228	0.3192	0.3156	0.3121
-0.3	0.3821	0.3783	0.3745	0.3707	0.3669	0.3632	0.3594	0.3557	0.3520	0.3483
-0.2	0.4207	0.4168	0.4129	0.4090	0.4052	0.4013	0.3974	0.3936	0.3897	0.3859
-0.1	0.4602	0.4562	0.4522	0.4483	0.4443	0.4404	0.4364	0.4325	0.4286	0.4247
-0.0	0.5000	0.4960	0.4920	0.4880	0.4840	0.4801	0.4761	0.4721	0.4681	0.4641
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9278	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990
3.1	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998

Selected Percentiles

Cumulative probability $\Phi(z)$:	.90	.95	.975	.98	.99	.995	.999
z:	1.282	1.645	1.960	2.054	2.326	2.576	3.090

REFERENCES

Ackoff, R. L. 1956. "The Development of Operations Research as a Science," *Operations Research* 4: 256.

Aggarwal, S. C. 1985. "MRP, JIT, OPT, FMS? Making Sense of Production Operations Systems," *Harvard Business Review*, September–October, pp. 8–16.

Anderson, J., R. Schroeder, S. Tupy, and E. White. 1982. "Material Requirements Planning Systems: The State-of-the-Art," *Production and Inventory Management* **23**(4): 51–67.

Arguello, M. 1994. "Review of Scheduling Software," Technology Transfer 93091822A-XFR, SEMATECH, Austin, TX.

Askin, R. G., and C. R. Standridge. 1993. *Modeling and Analysis of Manufacturing Systems*. New York: Wiley.

A. W. Shaw Company. 1915. *The Library of Factory Management*, vols. 1–5. Chicago: A. W. Shaw.

Axsäter, S. 1993. "Continuous Review Policies for Multi-Level Inventory Systems with Stochastic Demand," in *Handbooks in Operations Research and Management Science, vol. 4: Logistics of Production and Inventory*. S. C. Graves, A. H. G. Rinnooy Kan, and P. H. Zipkin (eds.). New York: North-Holland.

Babbage, C. 1832. *On the Economy of Machinery and Manufactures*. London: Charles Knight. Reprint, Augustus M. Kelley, New York, 1963.

Bahl, H. C., L. P. Ritzman, and J. N. D. Gupta. 1987. "Determining Lot Sizes and Resource Requirements: A Review," *Operations Research* **35**(3): 329–345.

Baker, K. R. 1993. "Requirements Planning," in *Handbooks in Operations Research and Management Science, vol 4: Logistics of Production and Inventory*, S. C. Graves, A. H. G. Rinnooy Kan, and P. H. Zipkin (eds.). New York: North-Holland.

Baker, W. M. 1994. "Understanding Activity-Based Costing," *Industrial Management*, March/April, **36**: 28–30.

Barnard, C. I. 1938. *The Functions of the Executive*. Cambridge, MA: Harvard University Press.

Barnes, R. 1937. *Motion and Time Study*. New York: Wiley.

Bartholdi, J. J., and D. D. Eisenstein. 1996. "A Production Line that Balances Itself," *Operations Resarch* **44**(1): 21–34.

Baumol, W. J., S. Blackman, and E. N. Wolff. 1989. *Productivity and American Leadership: The Long View*. Cambridge, MA: MIT Press.

Bazaraa, M. S., and C. M. Shetty. 1979. *Nonlinear Programming: Theory and Algorithms*. New York: Wiley.

Benjaafar, S., and M. Sheikhzadeh. 1997. "Scheduling Policies, Batch Sizes, and Manufacturing Lead Times," *IIE Transactions* **29**(2): 159–166.

Blackburn, J. D. (ed.). 1991. *Time-Based Competition: The Next Battleground in American Manufacturing*. Homewood, IL: Irwin.

Blackstone, J. H., Jr., D. T. Phillips, and G. L. Hogg. 1982. "A State-of-the-art Survey of Dispatching Rules for Manufacturing Job Shop Operations," *International Journal of Production Research* **20**(1): 27–45.

Bonneville, J. H. 1925. *Elements of Business Finance.* Englewood Cliffs, NJ: Prentice-Hall.

Boorstein, D. J. 1958. *The Americans: The Colonial Experience.* New York: Random House.

———. 1965. *The Americans: The National Experience.* New York: Random House.

———. 1973. *The Americans: The Democratic Experience.* New York: Random House.

Boudette, N. 1999. "Europe's SAP Scrambles to Stem Big Glitches—Software Giant to Tighten Its Watch After Snafus at Whirlpool, Hershey," *The Wall Street Journal,* Nov 4.

Bourland, K. 1992. "Spartan Industries," Case Study, Amos Tuck School, Dartmouth College.

Box, G. E. P., and G. M. Jenkins. 1970. *Time Series Analysis, Forecasting and Control.* San Francisco: Holden-Day.

Bradt, L. J. 1983. "The Automated Factory: Myth or Reality," *Engineering: Cornell Quarterly* 3(13).

Brown, R. G. 1967. *Decision Rules for Inventory Management.* New York: Holt, Rinehart and Winston.

Browne, J., J. Harhen, and J. Shivnan. 1988. *Production Management Systems.* Reading, MA: Addison-Wesley.

Bryant, K. L., and H. C. Dethloff. 1990. *A History of American Business.* Englewood Cliffs, NJ: Prentice-Hall.

Buzacott, J. A., and J. G. Shanthikumar. 1993. *Stochastic Models of Manufacturing Systems.* Englewood Cliffs, NJ: Prentice-Hall.

Carlier, J., and E. Pinson. 1988. "An Algorithm for Solving the Job-Shop Problem," *Management Science* **35:** 164–176.

Carnegie, A. 1920. *Autobiography of Andrew Carnegie.* Boston: Houghton Mifflin.

Cerveny, R. P., and L. W. Scott. 1989. "A Survey of MRP Implementation," *Production and Inventory Management* **30**(3): 31–34.

Chandler, Alfred D., Jr. 1977. *The Visible Hand: The Managerial Revolution in American Business.* Cambridge, MA: Belknap Press.

———. 1984. "The Emergence of Managerial Capitalism," *Business History Review* **58:** 473–503.

———. 1990. *Scale and Scope: The Dynamics of Industrial Capitalism.* Cambridge, MA: Harvard University Press.

Chandler, Alfred D., and S. Salsbury. 1971. *Pierre S. Du Pont and the Making of the Modern Corporation.* New York: Harper & Row.

Charney, C. 1991. *Time to Market: Reducing Product Lead Time.* Dearborn, MI: Society of Manufacturing Engineers.

Cherington, P. T. 1920. *The Elements of Marketing.* New York: Macmillan.

Churchman, C. W. 1968. *The Systems Approach.* New York: Dell.

Clark, A., and H. Scarf. 1960. "Optimal Policies for a Multi-Echelon Inventory Problem," *Management Science* **36:** 1329–1338.

Clark, K. B., R. H. Hayes, and C. Lorenz. 1985. *The Uneasy Alliance: Managing the Productivity-Technology Dilemma.* Boston: Harvard Business School Press.

Cohen, M. A., Y. S. Zheng, and V. Agrawal. 1994. "Service Parts Logistics Benchmark Study." Working paper, Wharton School, University of Pennsylvania, Philadelphia.

Cohen, S. S., and J. Zysman. 1987. *Manufacturing Matters: The Myth of the Post-Industrial Economy.* New York: Basic Books.

Consiglio, M. 1969. "Leonardo da Vinci: The First IE?" *Industrial Engineering* **1:** 71.

Copley, F. B. 1923. *Frederick W. Taylor: Father of Scientific Management.* New York: Harper and Brothers.

Cray, E. 1979. *Chrome Colossus: General Motors and Its Times.* New York: McGraw-Hill.

Crosby, P. B. 1979. *Quality Is Free: The Art of Making Quality Certain.* New York: McGraw-Hill.

———. 1984. *Quality Without Tears: The Art of Hassle-Free Management.* New York: McGraw-Hill.

Daskin, M. S. 1995. *Network and Discrete Location.* New York: Wiley.

Davidson, K. M. 1990. "Do Megamergers Make Sense?" in *Mergers, Acquisitions and Leveraged Buyouts,* R. L. Kuhn (ed.) Homewood, IL: Irwin.

de Kok, T. 1993. "Back-Order Lead Time Behavior in (S,Q)-Inventory Models with Compound Renewal Demand." Working paper, School of Technology Management, Eindhoven University of Technology, Eindhoven, The Netherlands.

Deleersnyder, J. L., T. J. Hodgson, R. E. King, P. J. O'Grady, and A. Savva. 1992. "Integrating Kanban Type Pull Systems and MRP Type Push Systems: Insights from a Markovian Model," *IIE Transactions* **24**(3): 43–56.

Deming, W. E. 1950a. *Some Theory of Sampling.* New York: Wiley.

———. 1950b. *Elementary Principles of the Statistical Control of Quality.* Tokyo: Union of Japanese Science and Engineering.

———. 1960. *Sample Design in Business Research.* New York: Wiley.

———. 1982. *Quality Productivity and Competitive Position.* Cambridge, MA: Massachusetts Institute of Technology, Center for Advanced Engineering Study.

———. 1986. *Out of the Crisis.* Cambridge, MA: MIT Press.

Dertouzos, M. L., R. K. Lester, and R. M. Solow. 1989. *Made in America: Regaining the Productive Edge.* Cambridge, MA: MIT Press.

Deuermeyer, B. 1994. Interoffice Memorandum on Undergraduate Curriculum in Industrial Engineering, Texas A&M University.

Deuermeyer, B., and L. B. Schwarz. 1981. "A Model for the Analysis of System Service Level in Warehouse/Retailer Distribution Systems: The Identical Retailer Case." In: *Multi-Level Production/Inventory Control Systems: Theory and Practice.* L. B. Schwarz (ed.). Amsterdam: North-Holland, 163–193.

DeVor, R., T. Chang, and J. Sutherland. 1992. *Statistical Quality Design and Control: Contemporary Concepts and Methods.* New York: Macmillan.

Drucker, P. F. 1954. *The Practice of Management.* New York: Harper & Row.

Dudek, R. A., S. S. Panwalkar, and M. L. Smith. 1992. "The Lessons of Flowshop Scheduling Research," *Operations Research* **40**(1): 7–13.

Duncan, W. J. 1989. *Great Ideas in Management: Lessons from the Founders and Foundations of Managerial Practice.* San Francisco: Jossey-Bass Publishers.

Edmondson, G., and A. Reinhardt. 1997. "Silicon Valley on the Rhine," *Business Week,* November 3, 162–166.

Einstein, A. 1950. Quoted by Lincoln Barnett, "The Meaning of Einstein's New Theory," *Life,* January 9, 22.

Emerson, H. P., and D. C. E. Naehring. 1984. *Origins of Industrial Engineering.* Norcross, GA: Institute of Industrial Engineers.

Erlenkotter, D. 1989. "An Early Classic Misplaced: Ford W. Harris's Economic Order Quantity Model of 1915," *Management Science* **35**(7): 898–900.

———. 1990. "Ford Whitman Harris and the Economic Order Quantity Model," *Operations Research* **38**(6): 937–946.

Fayol, H. 1916. *Administration industrielle et générale,* Paris: Dunod. In English, *General and Industrial Management.* (Constance Storrs, trans.) London: Sir Isaac Pitman and Sons, 1949.

Federgruen, A. 1993. "Centralized Planning Models for Multi-Echelon Inventory Systems under Uncertainty." In *Handbooks in Operations Research and Management Science, vol. 4: Logistics of Production and Inventory,* S. C. Graves, A. H. G. Rinnooy Kan, and P. H. Zipkin (eds.). New York: North-Holland.

Federgruen, A., and Y. Zheng. 1992. "The Joint Replenishment Problem with General Joint Cost Structures," *Operations Research* **40:** 384–403.

——— and ———. 1992. "An Efficient Algorithm for Computing an Optimal (r, Q) Policy in Continuous Review Stochastic Inventory Systems," *Operations Research* **40:** 808–813.

Federgruen, A., and P. Zipkin. 1984. "Computational Issues in an Infinite Horizon, Multi-Echelon Inventory Model," *Operations Research* **32:** 818–836.

Feigenbaum, A. V. 1956. "Total Quality Control," *Harvard Business Review,* November.

———. 1961. *Total Quality Control: Engineering and Management.* New York: McGraw-Hill.

Feitzinger, E., and H. L. Lee. 1997. "Mass Customization at Hewlett-Packard: The Power of Postponement." *Harvard Business Review,* January–February, 116–121.

Fish, J. C. L. 1915. *Engineering Economics: First Principles.* New York: McGraw-Hill.

Fisher, M. L. 1997. "What Is the Right Supply Chain for Your Product?" *Harvard Business Review,* March–April, 105–116.

Flink, J. J. 1970. *America Adopts the Automobile, 1895–1910.* Cambridge, MA: MIT Press.

Follett, M. P. 1942. *Dynamic Administration: The Collected Papers of Mary Parker Follett.* H. C. Metcalf, and L. Urwick (eds.). New York: Harper.

Ford, H. 1926. *Today and Tomorrow.* New York: Doubleday. Reprint, Productivity Press, 1988.

Fordyce, J. M, and F. M. Webster. 1984. "The Wagner-Whitin Algorithm Made Simple." *Production and Inventory Management* **25**(2): 21–30.

Forrester, J. 1961. *Industrial Dynamics.* New York: MIT Press and Wiley.

Fourer, R., D. M. Gay, and B.W. Kernighan. 1993. *AMPL: A Modeling Language for Mathematical Programming.* San Francisco: Scientific Press.

Fox, R. E. 1980. "Keys to Successful Materials Management Systems: A Contrast Between Japan, Europe and the U.S." *23rd Annual Conference Proceedings,* APICS, 440–444.

Freidenfelds, J. 1981. *Capacity Extension: Simple Models and Applications.* Amsterdam: North-Holland.

Galbraith, J. K. 1958. *The Affluent Society.* Boston: Houghton Mifflin.

Garvin, D. 1988. *Managing Quality: The Strategic and Competitive Edge.* New York: Free Press.

Gilbreth, F. B. 1911. *Motion Study.* New York: Van Nostrand.

Gilbreth, F. B., and E. G. Gilbreth Carey. 1949. *Cheaper by the Dozen.* New York: T. Y. Crowell.

Gilbreth, L. M. 1914. *The Psychology of Management.* New York: Sturgis and Walton. Reprinted in W. R. Spriegel, and C. E. Myers (eds.). *The Writings of the Gilbreths.* Homewood IL: Irwin, 1953.

Glover, F. 1990. "Tabu Search: A Tutorial." *Interfaces* **20**(4): 79–94.

Goldratt, E. M., and J. Cox. 1984. *The Goal: A Process of Ongoing Improvement.* Croton-on-the-Hudson, NY: North River Press.

Goldratt, E. M., and R. E. Fox. 1986. *The Race.* Croton-on-the-Hudson, NY: North River Press.

Gordon, R. A., and J. E. Howell. 1959. *Higher Education for Business.* New York: Columbia University Press.

Gould, L. 1985. "Computers Run the Factory." *Electronics Week,* March 25.

Grant, E. L. 1930. *Principles of Engineering Economy.* New York: Ronald Press.

Grant, E. L., and R. Leavenworth. 1946. *Statistical Quality Control.* Milwaukee, WI: American Society for Quality Control.

Graves, S. C., A. H. G. Rinnooy Kan, and P. H. Zipkin (eds.). 1993. *Handbooks in Operations Research and Management Science, vol. 4: Logistics of Production and Inventory.* New York: North-Holland.

Gross, D., and C. Harris. 1985. *Fundamentals of Queueing Theory.* 2d ed. New York: Wiley.

Hackman, S. T., and R.C. Leachman. 1989. "A General Framework for Modeling Production," *Management Science* **35**(4): 478–495.

Hadley, G., and T. M. Whitin. 1963. *Analysis of Inventory Systems.* Englewood Cliffs, NJ: Prentice-Hall.

Hall, R. W. 1981. *Driving the Productivity Machine: Production Planning and Control in Japan.* Falls Church, VA: American Production and Inventory Control Society, Inc.

———. 1983. *Zero Inventories.* Homewood, IL: Dow Jones-Irwin.

Hammer, M., and J. Champy. 1993. *Reengineering the Corporation.* New York: HarperCollins.

Harris, F. W. 1913. "How Many Parts to Make at Once." *Factory: The Magazine of Management* **10**(2): 135–136, 152. Also reprinted in *Operations Research* **38**(6): 947–950, 1990.

Hausman, W. H., and N. K. Erkip. 1994. "Multi-Echelon vs. Single-Echelon Inventory Control Policies for Low-Demand Items." *Management Science* **40**: 597–602.

Hax, A. C., and D. Candea. 1984. *Production and Inventory Management.* Englewood Cliffs, NJ: Prentice-Hall.

Hayes, R. 1981. "Why Japanese Factories Work." *Harvard Business Review,* July–August, pp. 57–66.

Hayes, R., and S. Wheelwright. 1984. *Restoring Our Competitive Edge: Competing through Manufacturing.* New York: Wiley.

Hayes, R., S. Wheelwright, and K. Clark. 1988. *Dynamic Manufacturing: Creating the Learning Organization.* New York: Free Press.

Hitomi, K. 1979. *Manufacturing Systems Engineering.* London: Taylor and Francis.

Hodge, A. C., and J. O. McKinsey. 1921. *Principles of Accounting.* Chicago: University of Chicago Press.

Hopp, W. J., and M. L. Roof. 1998. "Quoting Manufacturing Due Dates Subject to a Service Level Constraint," Technical Report, Department of Industrial Engineering, Northwestern University, Evanston, IL.

Hopp, W. J., and M. L. Spearman. 1991. "Throughput of a Constant Work in Process Manufacturing Line Subject to Failures." *International Journal of Production Research* **29**(3): 635–655.

Hopp, W. J., and M. L. Spearman. 1993. "Setting Safety Leadtimes for Purchased Components in Assembly Systems." *IIE Transactions* **25**(2): 2–11.

Hopp, W. J., and M. L. Spearman, and I. Duenyas. 1993. "Economic Production Quotas for Pull Manufacturing Systems." *IIE Transactions* **25**(2): 71–79.

Hopp, W. J., and M. L. Spearman, and D. L. Woodruff. 1990. "Practical Strategies for Lead Time Reduction." *Manufacturing Review* **3**(2): 78–84.

Industrial Engineering. 1991. "Competition in Manufacturing Leads to MRP II." *Industrial Engineering,* **23**(7): 10–13.

Inman, R. A., and S. Mehra. 1990. "The Transferability of Just-in-Time Concepts to American Small Businesses." *Interfaces* **20:** 30–37, March–April.

Inman, R. R. 1993. "Inventory Is the *Flower* of All Evil." *Production and Inventory Management Journal* **34**(4): 41–45.

Jackson, P. L., W. L. Maxwell, and J. A. Muckstadt. 1985. "The Joint Replenishment Problem With a Powers of Two Restriction." *IIE Transactions* **17:** 25–32.

Jacobs, F. R. 1984. "OPT Uncovered: Many Production Planning and Scheduling Concepts Can Be Applied With or Without the Software." *Industrial Engineering,* October, 32–41.

Johnson, H. T., and R. S. Kaplan. 1987. *Relevance Lost: The Rise and Fall of Management Accounting.* Cambridge, MA: Harvard Business School Press.

Johnson, L. A., and D. C. Montgomery. 1974. *Operations Research in Production Planning, Scheduling, and Inventory Control.* New York: Wiley.

Johnson, S. M. 1954. "Optimal Two- and Three-Stage Production Schedules with Setup Times Included," *Naval Research Logistics Quarterly* **1:** 61–68.

Juran, J. M. 1964. *Managerial Breakthrough.* New York: McGraw-Hill.

———(ed.). 1988. *Juran's Quality Control Handbook,* 4th ed., F.M. Gryna (assoc. ed.). New York: McGraw-Hill.

———. 1989. *Juran on Leadership for Quality: An Executive Handbook.* New York: Free Press.

———. 1992. *Juran on Quality by Design: The New Steps for Planning Quality into Goods and Services.* New York: Free Press.

Kakar, S. 1970. *Frederick Taylor: A Study in Personality and Innovation.* Cambridge, MA: MIT Press.

Kanet, J. J. 1984. "Inventory Planning at Black & Decker." *Production and Inventory Management* **25**(3): 62–74.

———. 1988. "MRP 96: Time to Rethink Manufacturing Logistics." *Production and Inventory Management* **29**(2): 57–61.

Kaplan, R. S. 1986. "Must CIM Be Justified by Faith Alone?" *Harvard Business Review,* March–April, 87–95.

Karmarkar, U. S. 1987. "Lot Sizes, Lead Times and In-Process Inventories." *Management Science* **33**(3): 409–423.

———. 1989. "Getting Control of Just-in-Time," *Harvard Business Review,* September–October, 122–131.

Kearns, D. T., and D. A. Nadler. 1992. *Prophets in the Dark: How Xerox Reinvented Itself and Beat Back the Japanese.* New York: HarperCollins.

Kellermann, A. L., F. P. Rivara, N. B. Rushforth, J. G. Banton, D. T. Reay, J. T. Francisco, A. B. Locci, J. Prodzinski, B. B. Hackman, and G. Somes. 1993. "Gun Ownership as a Risk Factor for Homicide in the Home." *New England Journal of Medicine,* **329**(15): 1084–1091.

Kilbridge, M. D., and L. Wester. 1961. "A Heuristic Method of Assembly Line Balancing," *Journal of Industrial Engineering*. **12**(4): 292–298.

Klein, J. A. 1989. "The Human Costs of Manufacturing Reform." *Harvard Business Review,* March–April, pp. 60–66.

Kleinrock, L. 1975. *Queueing Systems,* vol. I: *Theory.* New York: Wiley.

Krajewski, L. J., B. E. King, L. P. Ritzman, and D. S. Wong. 1987. "Kanban, MRP, and Shaping the Manufacturing Environment." *Management Science* **33**(1): 39–57.

Kuhn, T. S. 1970. *The Structure of Scientific Revolutions.* Chicago: University of Chicago Press.

LaForge, R., and V. Sturr. 1986. "MRP Practices in a Random Sample of Manufacturing Firms," *Production and Inventory Management* **28**(3): 129–137.

Lamm, R. D. 1988. "Crisis: The Uncompetitive Society." In *Global Competitiveness.* M. K. Starr (ed.). New York: Norton.

Lee, H. L., and C. Billington. 1992. "Managing Supply Chain Inventory: Pitfalls and Opportunities," *Sloan Management Review* **33:** 65–73.

———. 1995. "The Evolution of Supply-Chain-Management Models and Practice at Hewlett-Packard." *Interfaces* **25**(5): 42–63.

Lee, H. L., C. Billington, and B. Carter. 1993. "Hewlett-Packard Gains Control of Inventory and Service through Design for Localization." *Interfaces* **23**(4): 1–20.

Lee, H. L., V. Padmanabhan, and S. Whang. 1997a. "The Bullwhip Effect in Supply Chains." *Sloan Management Review* **38**(3): 93–102.

———. 1997b. "Information Distortion in a Supply Chain: The Bullwhip Effect." *Management Science* **43**(4): 546–558.

Little, J. D. C. 1992. "Tautologies, Models and Theories: Can We Find 'Laws' of Manufacturing?" *IIE Transactions* **24:** 7–13.

Lough, W. H. 1920. *Business Finance.* New York: Ronald Press.

Lundrigan, R. 1986. "What's This Thing Called OPT?" *Production and Inventory Management* **27**(2): 2–12.

Maddison, A. 1984. "Comparative Analysis of the Productivity Situation in the Advanced Capitalist Countries." In *International Comparisons of Productivity and Causes of the Slowdown.* J. W. Kendrick (ed.). Cambridge, MA: Ballinger.

Majone, G. 1985. "Systems Analysis: A Genetic Approach." In *Handbook of Systems Analysis: Overview of Uses, Procedures, Applications, and Practice,* chapter 2. Hugh J. Misner and Edward S. Quade (eds.). New York: Elsevier.

Marion, J. B. 1970. *Classical Dynamics of Particles and Systems,* 2d ed. New York: Academic Press, 266.

Martino, J. P. 1983. *Technological Forecasting for Decision Making,* 2d ed. New York: North-Holland.

Maslow, A. 1954. *Motivation and Personality.* New York: Harper.

Mayo, E. 1933. *The Human Problems of an Industrial Civilization.* New York: Macmillan.

———. 1945. *The Social Problems of an Industrial Civilization.* Cambridge, MA: Division of Research, Graduate School of Business Administration, Harvard University.

McClain, J. O., and L. J. Thomas. 1985. *Operations Management: Production of Goods and Services,* 2d ed. Englewood Cliffs, NJ: Prentice-Hall.

McCloskey, J. F. 1987a. "The Beginnings of Operations Research 1934–1941." *Operations Research* **35**(1): 143–152.

———. 1987b. "British Operational Research in World War II." *Operations Research* **35**(3): 453–470.

———. 1987c. "U.S. Operations Research in World War II." *Operations Research* **35**(6): 910–925.

McGregor, D. 1960. *The Human Side of Enterprise.* New York: McGraw-Hill.

Michel, R. 1997. "Reinvention Reigns: ERP Vendors Redefine Value, Planning, and Elevate Customer Service." *Manufacturing Systems,* July, 28.

Micklethwait, J., and A. Woolridge. 1996. *The Witch Doctors.* New York: Random House.

Miller, J. G., and T. E. Vollmann. 1985. "The Hidden Factory." *Harvard Business Review,* September–October, 142–150.

Miser, H. J., and E. S. Quade (eds.). 1985. *Handbook of Systems Analysis: Overview of Uses, Procedures, Applications, and Practice.* New York: North-Holland.

——— (eds.). 1988. *Handbook of Systems Analysis: Craft Issues and Procedural Choices.* New York: North-Holland.

Mitchell, W. N. 1931. *Production Management.* Chicago: University of Chicago Press.

Monden, Y. 1983. *Toyota Production System: Practical Approach to Production Management.* Norcross, GA: Industrial Engineering and Management Press.

Montgomery, D. C. 1991. *Introduction to Statistical Quality Control,* 2d ed. New York: Wiley.

Morton, T. E., and D. W. Pentico. 1993. *Heuristic Scheduling Systems with Applications to Production Systems and Project Management.* New York: Wiley.

Muckstadt, J. A., and L. J. Thomas. 1980. "Are Multi-Echelon Inventory Methods Worth Implementing in Systems with Low Demand Rates?" *Management Science* **26:** 483–494.

Muhs, W. F., C. D. Wrege, and A. Murtuza. 1981. "Extracts from Chordal's Letters: Pre-Taylor Shop Management." *Proceedings of the Academy of Management,* 41st annual meeting, San Diego, CA.

Mumford, L. 1943. *Technics and Civilization.* New York: Harcourt and Brace.

Munsterberg, H. 1913. *Psychology and Industrial Efficiency.* Boston: Houghton Mifflin.

Myers, F. S. 1990. "Japan's Henry Ford." *Scientific American* **262**(5): 98.

Nahmias, S. 1993. *Production and Operations Analysis.* 2d ed. Homewood, IL: Irwin.

Nahmias, S., and S. Smith. 1992. "Mathematical Models of Retailer Inventory Systems: A Review." In *Perspectives in Operations Management: Essays in Honor of Elwood S. Buffa.* R. K. Sarin (ed.). Boston: Kluwer.

Nellemann, D., and L. Smith. 1982. "Just-in-Time" vs. Just-in-Case Production/Inventory Systems Concepts Borrowed Back from Japan." *Production and Inventory Management,* second quarter, 12–21.

Nelson, D. 1990. *Frederick W. Taylor and the Rise of Scientific Management.* Madison: University of Wisconsin Press.

Niebel, B. 1993. *Motion and Time Study,* 9th ed. Homewood, IL: Irwin.

Ohno, T. 1988. *Toyota Production System: Beyond Large-Scale Production.* Cambridge, MA: Productivity Press (translation of *Toyota seisan hoshiki,* Tokyo: Diamond, 1978).

Ohno, T., and S. Mito. 1988. *Just-in-Time for Today and Tomorrow.* Cambridge, MA: Productivity Press (translation of *Naze hitsuyeo na mono o hitsuyeo na bun dake hitsuyeo na toki ni teikyeo shinai no ka,* Tokyo: Diamond, 1986).

Orlicky, J. 1975. *Material Requirements Planning: The New Way of Life in Production and Inventory Management.* New York: McGraw-Hill.

Parker, K. 1997. "The Great Trek Begins: Mid-sized Manufacturers Migrate to Client/Server Enterprise Systems," *Manufacturing Systems,* January.

Peterson, R., and E. A. Silver. 1985. *Decision Systems for Inventory Management and Production Planning.* 2d ed., New York: Wiley.

Pierson, F. C., et al. 1959. *The Education of American Businessmen.* New York: McGraw-Hill.

Pinedo, M. 1995. *Scheduling: Theory, Algorithms, and Systems.* Englewood Cliffs, NJ: Prentice-Hall.

Pinedo, M., and X. Chao. 1999. *Operations Scheduling: With Applications in Manufacturing and Services.* Boston: Irwin/McGraw-Hill.

Plossl, G. W. 1985. *Production and Inventory Control,* 2d ed. Englewood Cliffs, NJ: Prentice-Hall.

Pollard, H. R. 1974. *Developments in Management Thought.* London: Heinemann.

Polya, G. 1954. *Patterns of Plausible Inference.* Princeton, NJ: Princeton University Press.

Popper, K. 1963. *Conjectures and Refutations.* London: Routledge & Kegan Paul Ltd.

Rao, A. 1989. "A Survey of MRP-II Software Suppliers' Trends in Support of Just-in-Time." *Production and Inventory Management,* third quarter, 14–17.

Ravenscraft, D. J., and F. M. Scherer. 1987. *Mergers, Sell-Offs, and Economic Efficiency.* Washington: Brookings Institute.

Raymond, F. E. 1931. *Quantity and Economy in Manufacture.* New York: McGraw-Hill.

Roderick, L. M., D. T. Phillips, and G. L. Hogg, 1991. "A Comparison of Order Release Strategies in Production Control Systems." *International Journal of Production Research* **30**(2): 1991.

Roethlisberger, F. J., and W. J. Dickson. 1939. *Management and the Worker.* Cambridge, MA: Harvard University Press.

Roundy, R. 1985. "98% Effective Integer Ratio Lot-Sizing for One Warehouse Multi-Retailer Systems." *Management Science* **31:** 1416–1430.

———. 1986. "98% Effective Lot-Sizing Rule for Multi-Product, Multi-Stage Production Inventory Systems." *Mathematics of Operations Research* **11:** 699–727.

Sage, A. P. 1992. *Systems Engineering.* New York: Wiley.

Sanderson, R. J., J. A. Cambell, and J. D. Meyer. 1982. *Industrial Robots, A Summary and Forecast for Manufacturing Managers.* Lake Geneva, WI: Tech Tran Corporation.

Scherer, F. M., and D. Ross. 1990. *Industrial Market Structure and Economic Performance,* 3d ed. Boston: Houghton Mifflin.

Schmenner, R. W. 1993. *Production/Operations Management: From the Inside Out,* 5th ed. New York: Macmillian.

Schonberger, R. J. 1982. *Japanese Manufacturing Techniques: Nine Hidden Lessons in Simplicity.* New York: Free Press.

———. 1986. *World Class Manufacturing: The Lessons of Simplicity Applied.* New York: Free Press.

———. 1990. *Building a Chain of Customers: Linking Business Functions to Create a World Class Company.* New York: Free Press.

Schroeder, R., J. Anderson, S. Tupy, and E. White. 1981. "A Study of MRP Benefits and Costs." *Journal of Operations Management* **2**(1): 1–9.

Schumacher, B. G. 1986. *On the Origin and Nature of Management.* Norman, OK: Eugnosis.

Schwarz, L. B. (ed.). 1981. *Multi-Level Production/Inventory Control Systems: Theory and Practice.* Amsterdam: North-Holland.

———. 1998. "A New Teaching Paradigm: The Information/Control/Buffer Portfolio." *Production and Operations Management* **7**(2): 125–131, summer.

Scott, W. D. 1913. *Increasing Human Efficiency in Business.* New York: Macmillan.

Sethi, K. S., S. P. Sethi. 1990. "Flexibility in Manufacturing: A Survey," *International Journal of Flexible Manufacturing Systems* **2,** 289–328.

Shafritz, J. M., and J. S. Ott. 1992. *Classics of Organization Theory,* 3d ed., Pacific Grove, CA: Brooks/Cole Publishing Company.

Sherbrooke, C. C. 1992. *Optimal Inventory Modeling of Systems: Multi-Echelon Techniques.* New York: Wiley.

Shewhart, W. A. 1931. *Economic Control of Quality of Manufactured Product.* New York: Van Nostrand.

Shingo, S. 1985. *A Revolution in Manufacturing: The SMED System.* Cambridge, MA: Productivity Press.

———. 1986. *Zero Quality Control: Source Inspection and the Poka-Yoke System.* Cambridge, MA: Productivity Press.

———. 1989. *A Study of the Toyota Production System from an Industrial Engineering Viewpoint.* Cambridge, MA: Productivity Press.

———. 1990. *Modern Approaches to Manufacturing Improvement: The Shingo System.* A. Robinson (ed.). Cambridge, MA: Productivity Press.

Silver, A., D. Pyke, and R. Peterson. 1998. *Inventory Management and Production Planning and Scheduling.* New York: Wiley.

Simchi-Levi, D., P. Kaminsky, and E. Simchi-Levi. 1999. *Designing and Managing the Supply Chain: Concepts, Strategies and Cases.* Burr Ridge, IL: Irwin/McGraw-Hill.

Simons, Jr., J. V., and W. P. Simpson III. 1997. "An Exposition of Multiple Constraint Scheduling as Implemented in the Goal System (Formerly Disaster)." *Production and Operations Management* **6**(1): 3–22.

Singer, C., E. Flomyard, A. Hall, and T. Williams. 1958. *A History of Technology*. Oxford: Clarendon Press.

Skinner, W. 1969. "Manufacturing—The Missing Link in Corporate Strategy." *Harvard Business Review*, May/June, 156.

———. 1974. "The Focused Factory." *Harvard Business Review*, May–June, 113-121.

———. 1985. *Manufacturing: The Formidable Competitive Weapon*. New York: Wiley.

———. 1985b. "The Taming of Lions: How Manufacturing Leadership Evolved, 1780–1984." In K. B. Clark, R. H. Hayes, and C. Lorenz, *The Uneasy Alliance: Managing the Productivity-Technology Dilemma*, Boston: Harvard University Press.

———. 1986. "The Productivity Paradox." *Harvard Business Review*, July–August, 55–59.

———. 1988. "What Matters to Manufacturing." *Harvard Business Review*, January–February, 10–16.

Smith, A. 1776. *An Inquiry into the Nature and Causes of the Wealth of Nations*. Chicago: Great Books of the Western World, vol. 39, *Encyclopaedia Britannica*, 1952.

Spearman, M. L. 1991. "An Analytic Congestion Model for Closed Production Systems with IFR Processing Times," *Management Science* **37**(8): 1015–1029.

Spearman, M. L., W.J. Hopp, and D. L. Woodruff. 1989. "A Hierarchical Control Architecture for CONWIP Production Systems." *Journal of Manufacturing and Operations Management* **2**: 147–171.

Spearman, M. L., and S. Kröckel. 1999. "Batch Sizing to Minimize Flow Times in a Multi-Product System with Significant Changeover Times." Technical Report. Atlanta: Georgia Institute of Technology.

Spearman, M. L., D. L. Woodruff, and W. J. Hopp. 1989. "CONWIP: A Pull Alternative to Kanban." *International Journal of Production Research* **28**(5): 879–894.

Spearman, M. L., and M. A. Zazanis. 1992. "Push and Pull Production Systems: Issues and Comparisions." *Operations Research* **40**(3): 521–532.

Spearman, M. L., and R. Q. Zhang. 1999. "Optimal Lead Time Policies." *Management Science* **45**(2): 290–295.

Spriegel, W. R., and C. E. Myers (eds.). 1953. *The Writings of the Gilbreths*. Homewood, IL: Irwin.

Stalk, G., and T. M. Hout. 1990. *Competing Against Time: How Time-Based Competition Is Reshaping Global Markets*. New York: Free Press.

Stedman, C. 1999. "Survey: ERP Costs More Than Measurable ROI," *Computerworld*, April 5.

Sterman, J. D. 1989. "Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment." *Management Science* **35**(3): 321–339.

Stover, John, F. 1961. *American Railroads*. Chicago: University of Chicago Press.

Suri, R. 1998. *Quick Response Manufacturing: A Companywide Approach to Reducing Leadtimes*. Portland, OR: Productivity Press.

Suri, R., and S. de Treville. 1992. "Time Is Money." *OR/MS Today*, October.

———. 1993. "Rapid Modeling: The Use of Queueing Models to Support Time-Based Competitive Manufacturing." In *Operations Research in Production Planning and Control*. G. Fandel, T. Gulledge, and A. Jones (eds.). New York: Springer-Verlag.

Suri, R., J. L. Sanders, and M. Kamanth. 1993. "Performance Evaluation of Production Networks." In *Handbooks in Operations Research and Management Science, vol 4: Logistics of Production and Inventory*. S. C. Graves, A. H. G. Rinnooy Kan, and P. H. Zipkin (eds.). New York: North-Holland.

Svoronos, A., and P. Zipkin. 1988. "Estimating the Performance of Multi-Level Inventory Systems." *Operations Research* **36**: 57–72.

Tardif, V. 1995. "Detecting Scheduling Infeasibilities in Multi-Stage, Finite Capacity, Production Environments," Ph.D. dissertation, Northwestern University, Evanston, IL.

Taft, E. W. 1918. "Formulas for Exact and Approximate Evaluation—Handling Cost of Jigs and Interest Charges of Product Manufactured Included." *The Iron Age* **101**: 1410–1412.

Taylor, A. 1997. "How Toyota Defies Gravity." *Fortune,* December 8, 100–108.

Taylor, F. W. 1903. "Shop Management." *Transactions of the ASME* **24:** 1337–1480.

———. 1911. *The Principles of Scientific Management.* New York: Harper & Row.

Thomas, P. R. 1990. *Competitiveness Through Total Cycle Time: An Overview for CEO's.* New York: McGraw-Hill.

———. 1991. *Getting Competitive: Middle Managers and the Cycle Time Ethic.* New York: McGraw-Hill.

Thompkins, J. A., and J. A. White. 1984. *Facilities Planning.* New York: Wiley.

Thompson, J. R., and J. Koronacki. 1992. *Statistical Process Control for Quality.* New York: Chapman & Hall.

Thompson, M. B. 1992. "Why Finite Capacity?" *APICS—The Performance Advantage,* June, 50–54.

Towne, H. R. 1886. "The Engineer as an Economist." *ASME Transactions* **7:** 428–432.

Turino, J. 1992. *Managing Concurrent Engineering Buying Time to Market.* New York: Van Nostrand Reinhold.

U.S. Department of Commerce. 1972. *Statistical Abstract of the United States.* 93d annual edition, Bureau of the Census.

———. 1977. *Statistical Abstract of the United States.* Economics and Statistics Administration, Bureau of the Census, Table 664, 842 and 758.

Ure, A. 1835. *The Philosophy of Manufactures: Or an Exposition of the Scientific, Moral and Commercial Economy of the Factory System of Great Britain.* London: Charles Knight. Reprint, Augustus M. Kelley, New York, 1967.

Urwick, L. 1947. *The Elements of Administration.* London: Pitman.

Vollmann, T. E., W. L. Berry, and D. C. Whybark. 1992. *Manufacturing Planning and Control Systems,* 3d ed., Burr Ridge, IL: Irwin.

Wack, P. 1985. "Scenarios: Uncharted Waters Ahead." *Harvard Business Review,* September–October, 73–89.

Wagner, H. M., and T. M. Whitin. 1958. "Dynamic Version of the Economic Lot Size Model." *Management Science* **5**(1): 89–96.

Waring, S. P. 1991. *Taylorism Transformed: Scientific Management Theory Since 1945.* Chapel Hill: University of North Carolina Press.

Wellington, A. M. 1877. *The Economic Theory of the Location of Railways.* New York: Wiley.

Wheelwright, S. 1981. "Japan—Where Operations Really Are Strategic." *Harvard Business Review,* July–August, 67–74.

Whiteside, D., and J. Arbose. 1984. "Unsnarling Industrial Production: Why Top Management Is Starting to Care," *International Management,* March, 20–26.

Whitin, T. M. 1953. *The Theory of Inventory Management.* Princeton, NJ: Princeton University Press.

Whitt, W. 1983. "The Queueing Network Analyzer." *Bell System Technology Journal* **62**(9): 2779–2815.

———. 1993. "Approximating the GI/G/m Queue." *Production and Operations Management,* **2**(2): 114–161.

Wight, O. 1970. "Input/Output Control: A Real Handle on Lead Time." *Production and Inventory Management Journal* **11**(3): 9–31.

———. 1974. *Production and Inventory Management in the Computer Age.* Boston: Cahners Books.

———. 1981. *MRP II: Unlocking America's Productivity Potential.* Boston: CBI Publishing.

Wilson, B. 1984. *Systems: Concepts, Methodologies, and Applications.* New York: Wiley.

Wilson, R. H. 1934. "A Scientific Routine for Stock Control." *Harvard Business Review* **13**(1): 116–128.

Winters, P. 1960. "Forecasting Sales by Exponentially Weighted Moving Averages." *Management Science* **6:** 324–342.

Woodruff, D., and M. Spearman. 1992. "Sequencing and Batching for Two Classes of Jobs with Deadlines and Setup Times," *Journal of Production and Operations Management,* **1:** 87–102.

Wrege, C. D., and R. G. Greenwood. 1991. *Frederick W. Taylor—The Father of Scientific Management: Myth and Reality.* Homewood, IL: Irwin.

Wren, D. 1987. *The Evolution of Management Thought.* 3d ed. New York: Wiley

Yates, R. 1992. "On the Road with the 'Messiah of Management' as He Tries to Do for His Country What He Did for Japan." *Chicago Tribune,* February 16, Section 10, 16.

Zais, A. 1986. "IBM Reigns in Dynamic MRP II Marketplace." *Computerworld,* January 27.

Zipkin, P. H. 1986. "Inventory Service-Level Measures: Convexity and Approximation." *Management Science* **32:** 975–981.

———. 1991. "Does Manufacturing Need a JIT Revolution?" *Harvard Business Review,* January–February, 40–50.

———. 1999. *Foundations of Inventory Management.* New York: McGraw-Hill.

INDEX

Notation

General Conventions:

- A subscript "a" indicates a parameter that describes interarrival times to a station. For example, t_a represents the average time between arrivals to a station or line.
- A subscript "e" indicates a parameter that describes "effective" process times at a station. For example, t_e represents the average process time at a station including detractors such as downtime, setups, yield loss, etc.
- A parameter followed by (i) indicates that the parameter applies to station i, as in TH(i), CT(i), $t_e(i)$, $c_e(i)$, and so on.
- A superscript * indicates a parameter that describes an "ideal" system without detractors. For example, r_b^* and T_0^* are the bottleneck rate and raw process time for a line with no downtime, setups, yield loss, or other inefficiencies.
- A superscript "P" indicates a parameter that describes a "practical" system. For example, r_b^P and T_0^P are the bottleneck rate and raw process time for a line operating under realistic conditions.

Mathematical Symbols:

CV coefficient of variation of a random variable, which is the standard deviation divided by the mean.

c_0 CV of natural (no detractors) process time at a station.

c_a CV of the time between arrivals to a station.

c_e CV of effective process time at a station.

c_d CV of the time between departures from a station.

CT$_q$ average queue time at a station. For single machine stations: CT$_q$ $= \left(\frac{c_a^2+c_e^2}{2}\right)\left(\frac{u}{1-u}\right) t_e$.

CT cycle time, which is measured as the average time from when a job is released into a station or line to when it exits. (Where ambiguity is possible cycle time at station i is written as CT(i).) Note that CT $=$ CT$_q + t_e$.

FGI finished goods inventory. For end items, FGI represents the store of final product waiting to be shipped to customers. For components, FGI can also represent "crib" inventory, which is stock in an intermediate location such as before an assembly operation.

LT lead time, a management constant indicating the time allotted for production of a part on a given routing.

r_e effective rate, or capacity, of a station.

r_b bottleneck rate of a line, defined as the rate of the station with the highest utilization.

RMI raw material inventory, consisting of the physical inputs at the start of a production process.

s service level. In make-to-order systems, s is measured as the fraction of jobs for which cycle time is less than or equal to lead time. In make-to-stock systems, s is measured as the fill rate, or fraction of demands that are filled from stock.

σ_0 standard deviation of natural (no detractors) process time at a station.

σ_e standard deviation of the effective process time at a station.

σ_{CT} standard deviation of the cycle time in a line.

TH throughout, measured as the average output of a production process (machine, station, line, plant) per unit time.

T_0 raw process time, which is the sum of the mean effective process times of the stations in a line.

t_0 average natural (no detractors) process time at a station.

t_a average time between arrivals to a line or station. At any station, $\text{TH} = 1/t_a$.

t_e mean effective process time (average time required to do one job) including all "detractors" such as setups, downtime, etc. It does not include time the station is starved for lack of work or blocked by busy downstream stations.

u utilization, defined as the fraction of time a station is not idle for lack of parts. $u = \text{TH}t_e/m$, where m is the number of parallel machines at the station.

WIP work in process, which consists of inventory between the start and end points of a routing.

WIP_q average WIP in queue at a station.

W_0 critical WIP level for a line, which is the WIP required for a line with no variability to achieve maximum throughput (r_b) with minimum cycle time (T_0). For a line with parameters, r_b and T_0, $W_0 = r_bT_0$.

[General Information]
书名=工厂物理学
作者=
页数=698
SS号=0
出版日期=

封面页
书名页
版权页
前言
目录
正文
跋