

第十四章 车间作业控制

千里之行，始于足下。

——老子

14.1 引言

车间作业控制 (Shop floor control, SFC) 是计划与部件的结合处 (where planning meets parts)。正因为如此，它是生产计划与控制体系的基础。又因为接近实际制造过程，SFC 也是收集其他计划与控制模块所用数据的自然工具。定义良好的 SFC 模块既控制通过工厂的物流，有使得生产计划体系的其他部分易于设计和管理。¹

SFC 在生产计划层级中有着逻辑重要性，但实践中人们往往不太关心它。部分原因是，它被过于狭隘地看作是纯粹的物料流动控制。在这种观点之下，似乎一旦拥有良好的排程，SFC 功能就可以通过连接到部件的路由选择 (routing slip) 以及给定要访问的加工中心的顺序来实现；工人仅仅按排程给定的顺序加工部件，然后依据路由选择移动它们。如我们在这里和第十五章所见，即使拥有非常有效的排程模块，对物料流动的控制也往往不是那么简单。任何的排程体系都不希望有随机扰动，但 SFC 模块无论如何都要适应它们。更进一步地，如我们已提起并将在本章中深入讨论，物料流动控制只是 SFC 的一个非常小的关注点。当考虑到 SFC 应有的其他功能，这个模块就在整个的计划层级中呈现非常重要的作用。

SFC 缺乏关注，可能还有另外一个原因。一系列来自运营管理文献的结果指出，对于工厂绩效而言，控制物料流动的决策没有塑造生产环境的决策重要。(453|454) Krajewski 等 (1987) 使用仿真结果显示了，通过压缩准备时间、提高产出、提高劳动力柔性等措施改进生产环境获得的收益远大于从再定货点或 MRP 体系转到看板体系的收益。在其研究的基础上，他们推断 (1) 改造生产环境是日本的成功案例的关键，(2) 如果企业充分地改善了生产环境，那么它用那种生产控制体系都没有什么差别。在更加具体的点上，Roderick 等 (1991) 使用仿真来显示，投料速率对绩效的影响远大于单个机器处的作业排程。他们的结论是，主生产计划 (MPS) 平滑很可能比控制产线上工件的精巧技术有着更大的有益影响。

如果将 SFC 狭隘地理解为分派或机器之间的物流控制，那么类似于上文的研究确实趋于最小化它的重要性。然而，如果采用更广泛的观点，认识到 SFC 控制物流以及建立与其他功能之间的联系，那么 SFC 模块的设计有益于塑造生产环境。例如，安装看板体系的决定就显示了对小批制造和压缩准备时间的承诺。此外，拉式系统自动控制工厂的投料速率，因而能够实现 Roderick 等定义的关键收益。

可是看板 (或其他类似的) 能真正实现这些环境改善吗？Krajewski 等暗示环境改善，如压缩准备时间，在没有看板时也同样有效，而 JIT 支持者主张在推进这些改善时需要看板来提供必要的压力。我们的观点比较接近与 JIT 支持者；没有 SFC 促进环境改善的模块以及通过收集数据、记录它们的有效性，就非常难以定义杠杆区域以及做出稳固的变化。因此，我们将把改造生产环境视为 SFC 模块设计的部分内容。

¹ 我们提醒读者，这里称的模块包括与某个特定的计划与控制体系相关的决策制定、记录保持以及计算等所有内容。所以尽管 SFC 模块可能通过计算机程序来实现，但它不仅于此。实际上，某些 SFC 模块甚至根本无法计算机化。

在第四、十、十三章讨论的基础上，我们觉得最有效（也是易于管理）的生产环境是由拉式系统建立的。回忆起推式与拉式系统最基本的区别在于，推式系统计划生产而拉式系统授权生产。任何授权生产的拉式机制的本质在于约束产线中总库存的 **WIP** 上限。在我们看来，若没有建立 **WIP** 上限，这个系统不能被称为拉（*pull*）。补足这个定义还需要一连串的其他支持特征，包括压缩准备时间、员工交叉训练、单元布置、源头质量等等。这些技术应用的方式和程度取决于具体的系统。**SFC** 模块的目标是建立与第四章、第十章中检视的理想环境尽可能接近的实际环境。同时，**SFC** 模块应相对易于使用，与其他计划功能良好整合，有足够的柔性来适应工厂可能面对的变化。如我们将看到的，由于制造条件的设定区别很大，我们能这样做的程度也很不相同，而适当的 **SFC** 模块的本质也是如此。

图 14.1 显示了可以整合进入 **SFC** 模块的功能序列。这些功能的核心是**物料流动控制**（**material flow control, MFC**），没有它，**SFC** 就不是车间作业控制。物料流动控制是我们据以决定哪些加工任务投入工厂，哪些部件在各个工站处加工，以及哪些物料在工站之间移动的机制。尽管 **SFC** 有时被狭隘地理解为只包括物料流动控制，它还包括一些与物料流动控制紧密相关的其他功能，良好的 **SFC** 模块可以为它们提供平台。（454|455）

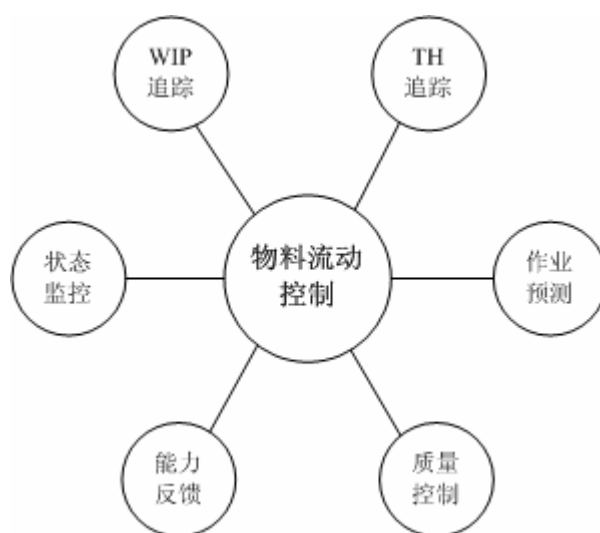


图 14.1 **SFC** 模块的潜在功能

WIP 追踪（**WIP tracking**）、**状态监控**（**status monitoring**）与**产出追踪**（**throughput tracking**）实时处理工厂事务。**WIP 追踪**包括识别产线中部件的当前位置。它的执行可以很详细以及自动化（如，通过光电扫描仪），或者很粗糙以及手工记录（如，在产线特定点处的日志）。**状态监控**指的是对 **WIP** 位置之外的其他参数，如机器状态（即，开机或关机）、班次情况，进行监控。**产出追踪**针对一个建立的生产定额和/或客户交期测量产线或工厂的产出，并可用于预测超时或变班的需要。

由于 **SFC** 模块处于执行实时控制决策的位置，它也很自然地处于实时监控这些类型的变化的位置。如果 **SFC** 模块在计算机上实现，这些数据收集与显示任务很可能与 **SFC** 模块的物料流动控制共享文件。即使物料流动控制由人工系统实现，也有必要将对系统的监视与控制协力，因为这可能对文书工作的形式产生影响。一种监控系统的具体机制称为**统计产出控制**（**statistical throughput control, STC**），我们在其中追踪完成时段性生产定额的过程。**STC** 的细节在 14.5.1 节给出。

除了收集实时状态的信息，**SFC** 模块也有益于收集、加工超出实时的与未来相关的信息。一种可能性是**实时仿真**（**real-time simulation**）功能，对产线中各点处特定部件的到达时机

做出投影。第十三章以离线活动的方式说明这个功能。然而，也可以将一个版本的实时仿真模块直接整合进入 SFC 模块。基本机制是，使用 WIP 追踪功能收集的当前 WIP 位置信息，加上一个物料流动的模型（如，基于传送带模型）来预测某个加工任务何时到达某个工站。能够从系统得到这些信息，使得产线人员可以预期和准备加工任务。

SFC 模块的一个稍显不同的功能是收集数据来更新产能预测。（455|456）这个**能力反馈（capacity feedback）**功能对于确保第十三章中讲到的高层级的计划模块与低层级的执行模块一致是非常重要的。由于 SFC 模块管制通过工厂的物料运动，它很自然地成为测度产出的位置。通过随时间向前监控输入，我们可以估计产线或工厂的实际产能。实现这一点的细节将在第 14.5.2 节讨论。

移动点代表了品质保证的天然机会，这个事实建立了 SFC 模块与**质量控制（quality control）**之间的联系。如果下游工站的操作员有权拒绝上游工站流出的不良工件，则 SFC 模块必须识别这项事物处理请求带来的扰动。物料流动控制功能必须认识到需要替换被拒绝的工件或者重工将带来工件到达的延迟；WIP 追踪功能必须注意到这些工件并未如预期般移动；作业预测功能必须考虑这些延迟从而组织作业投射。此外，由于质量问题必须放在控制的目标下考虑，所以常常能很方便地用这个系统对其保持记录。这些记录提供了与监控质量绩效以及识别改善机会的统计过程控制（SPC）之间的联系。

在本章的剩余部分，我们给出

1. 设计 SFC 模块之前必须解决的问题的综览。
2. CONWIP 作为 SFC 模块基础的讨论。
3. CONWIP 主题的扩展。
4. 机制，包括短期内追踪生产从而测度向定额的进程，以及长期内为其他计划模块收集、检验产能数据。

14.2 一般的考虑

人们很自然地想到通过说明控制机制自身的问题来开始对设计 SFC 系统的讨论，这些问题有：应当由计算机控制作业投放吗？应当使用看板卡片吗？等等。然而，更基本的问题需要首先解决。它们关系到 SFC 系统运行必须处于的一般物理和逻辑环境。

为了发展一个揭示 SFC 模块管理含义的合理视角，兼顾设计和控制两点来考虑车间作业控制很重要。设计解决建立一个可供决策的系统的问题，而控制解决决策本身的问题。例如，选择一种作业投放机制是个设计决策，而选择参数（如，WIP 水平）使这个机制工作就属于控制问题了。本章始于说明相对高层次的设计主题，然后逐渐移到低层次的控制主题。

14.2.1 总产能控制

生产控制体系在稳定环境下运行得最好。如 Krajewski 等（1987）的仿真研究显示，当需求稳定，产品组合恒定，加工执行良好时几乎任何类型的系统（如，再定货点、MRP 或看板）都能良好运行。（456|457）从制造的角度来看，我们想要建立产线并在在良好、节奏稳定、无扰动的条件下运行。事实上，在很大程度来说，这正是强调生产平滑和压缩换模时间的 JIT 竭力去做的。但创造一个平滑、简易的生产环境可能与赚钱、增长、保持市场份额以及确保长期生存能力的业务目标相冲突。客户需求波动、产品形成与衰退以及技术竞争促使我们依赖新的和不稳定的制程。因此，在寻求使环境稳定的机会时，我们必须小心在热心推动时忘记较高层级的目标。我们不应当仅仅因为原有技术比较稳定和易于管理，就放弃一

个通过新技术来获取策略优势的机会。

甚至当我们对市场需求做出反应时，也可以做些事情来避免工厂内部不必要的波动。一种稳定 SFC 运行环境的方法是，使用总产能控制来确保产线运行时接近最优负荷。目标是通过控制产线使用的全部或部分时间来避免产线速度的剧烈摇摆。总产能控制的具体选择包括

1. 改变班次的数目。例如，每天三班可用于重大需求时段，而每天只排两班可用于较轻的需求时段。工厂可以通过这个选择将产能与需求的季节波动匹配。然而，它一般涉及解雇和重新雇用工人，所以仅仅适用于长期需求变动（如，数月或更长）。

2. 改变每周工作日的数目。例如，周末可用于满足需求的激增。由于周末工人可以得到加班费，工厂可用这种方法实现比改变班次更小的影响。注意我们在这里讨论的是计划的加班，其中周末由于重大需求而提前计划使用。它与第十三章中讨论的用于追赶落后于定额的紧急加班很不相同。

3. 改变每日工作小时的数目。计划的加班的另一个来源是延长工作日，例如，从八小时到十小时。

4. 改变班次的水平。在手工作业中，产能可以通过增加工人数目（如，从工厂的其他部分浮动工人过来，或雇用临时工）而提高。在多机工站，管理人员可以通过改变使用中的机器数量来改变产能，可能也需要班次的变化。

5. 使用外部供应商。维持工厂或产线的稳定负载的一种方法是，转移某个水平之外的作业到另外一家企业。理想状况下，它转移了至少部分需求变动性的负担给供应商（代工企业）。²

如同总产能控制的意思，这些活动只能以比较粗略的方式改变有效产能。班次必须一体增加并且不常移除（Shifts must be added whole and only infrequently removed）。依据工会规则或人事政策，周末加班必须是特定的数量（如，一天或半天）。通过浮动工人改变产能的选择受限于工人技能水平和工厂其他部分的载荷。（457|458）增加和减少临时工需要培训和其他费用，也降低了这个选择的柔性。供应合同可能会要求最小量和/或最大量的作业送达供应商，所以这种方法只能转移企业面临的部分需求变动性。此外，供应商的找寻和认证很费时，所以供应合同很可能长期不变。

上述选择除了局限性，它们，或其他方法，能至少是粗略地匹配产能是很重要的。产线负载的巨大变动将引起通过产线的巨大变动性，并将严重地降低绩效。看板或 CONWIP 需要相当稳定的速率驱动的产线。我们将讨论一种不能通过总产能控制实现这种稳定性的产线拉式方案。然而，任何系统都不能完全缓解高度变动需求的负面影响。

14.2.2 瓶颈规划

第二篇中我们强调产线速率最终决定于瓶颈，也就是最慢的作业。在第七章用于说明基本工厂动力学的简单产品、单路线产线中，瓶颈作业代表了产线的最大速率。这个速率只能在产线中的WIP被许可变大³时达到，如图 14.2 所示。

² 当然，谁也不能保证供应商能比本企业更好地使用变动的需求。而且，有能力做到的供应商也很可能为之收取额外的费用。所以，供应商可能是有用的，但绝不是万能药。

³ 究竟要多大，当然取决于产线中的变动性，如第九章所讲述的。

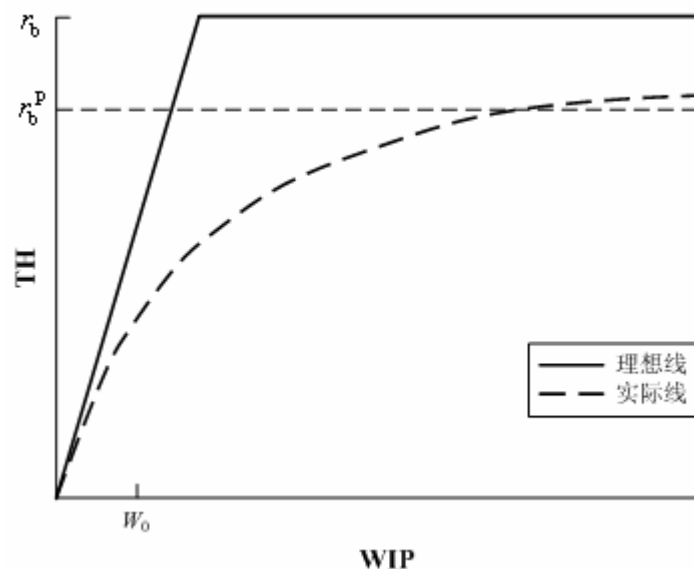


图 14.2 单产品产线中作为 WIP 函数的产出

在所有的工件遵循同样的路线，加工时间相同以至于某项作业对于所有的工件都是最慢的产线中，传送带模型就是对实际的精确表示，有益于分析以及形成直觉。在这种情况下，瓶颈对于产线绩效扮演着关键的角色，故而应由 SFC 模块给予特别关注。产出是瓶颈利用率的直接函数，于是很合理地想到依据瓶颈状态来触发向产线的投料。这种“由瓶颈处拉”的方案在某些系统中运行良好，我们稍后再对其进行讨论。

不管瓶颈在理论上有多重要，我们的经验是很少有制造企业能够有信心确认出瓶颈作业。原因在于，很少有制造环境接近单产品、单路线的产线。结果就是，一种产品的瓶颈机器可能不是另一种产品的瓶颈。（458|459）这可能引起瓶颈“漂移”，而具体取决于产品组合。回忆起图 10.9 显示的这种类型的行为，那个例子是机器 2 是产品 A 的瓶颈，机器 4 是产品 B 的瓶颈，而机器 3 则是产品 A 和 B 按 50-50 组合时的瓶颈。

多产品系统通常包含不同产品的不同路线。例如，图 14.3 显示了有一个共享工站的双路线系统。机器 3 是不是产品 A 的瓶颈取决于产品 B 的份量。类似地，产品 B 的瓶颈取决于产品 A 的份量。因此，这个系统的瓶颈也可以依据产品组合而漂移。此外，如果两条产品线分开管理，各条产线的瓶颈位置就在产线管理人员的控制之外。

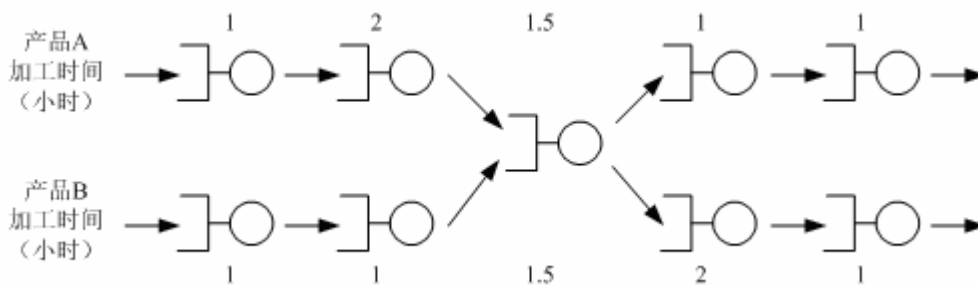


图 14.3 有共享资源的路线

上述讨论对 SFC 模块的设计有两条重要启示：

1. 稳定的瓶颈更易于管理。瓶颈界定清晰的产线比瓶颈复杂漂移的产线较易于建模

（即，使用传送带模型）和控制。管理人员可以聚焦于瓶颈状态，并在考虑产线其他部分时几乎只关注它们对瓶颈的影响（即，防止瓶颈的饥饿或阻塞）。如果碰巧是一条明确瓶颈的产线，我们就可以通过给予瓶颈良好处理，提供对其状态精确监控的 SFC 模块来利用这个优势。

2. 瓶颈可以被设计。尽管某些制造系统的瓶颈状况或多或少地受其他因素决定（如，所有关键制程的产能由于费用太高而不能改变），我们还是能够主动地影响瓶颈。例如，我们可以通过在某些工站处增加产能，确保它们在事实上永远不会限制产出，来降低潜在瓶颈的数量。在产能不贵的工站，这样做是合理的。⁴或者是相互作用的产线分划为单元；例如，图 14.3 中的产线可以通过增加一台额外的机器 3 来分划（或者在工站 3 是多机工站时将机器指派入产线）。这种类型的单元制造在工业中越来越流行，原因是大的部分变小了，而简单的单元比庞大、复杂的工厂易于管理。

尽管简化瓶颈行为的损益难以精确估计，很明显的是确实存在与复杂性相关的成本。要管理的最简单的工厂是路线独立，瓶颈清晰稳定的。对这种情况的任何偏离，都将提高变动性、拥堵以及低效率。这并不意味着我们应自动地增加产能直到类似于这种理想状况；要注意的仅仅是偏离它的动力。如果我们正被漂移的瓶颈烦恼，而它又可以通过不贵的产能来消除，那么增加产能就值得考虑。如果相互作用的产线能被分划开来并且不需要大的成本，那么它就值得研究。（459|460）

还有，产线设计与产能分配并不需要在整个工厂范围有效。指派一些量大的产品族到独立、设计良好的单元，而留下许多量小的产品族到工厂的低效的加工车间部分，有时候可以实现巨大的改进。这种“厂内之厂”的理念已被大量的研究者和实践者，最显著的要数 Wickham Skinner（1974），证明为**聚焦的工厂（focusd factory）**哲学的一部分。聚焦的工厂背后的主要观点是，工厂只能在一小部分事情上做得非常好，故而应当关注于一个小范围内的产品、制程、数量和市场。如我们将要在第三篇中反复见到的，简洁性在计划层级的各个水平，从低层的车间作业控制到长期的战略规划，都提供了重要的益处。

14.2.3 控制的跨度

第十三章中，我们已经讨论了将生产计划问题向较小、更易于管理的单元的拆分。我们将大部分问题沿着时间维度拆分，划入短期、中期和长期计划。但其他维度也可能是重要的。特别地，在大型工厂内有必要按产品或制程来划分，从而避免单条产线管理人员的过载。

控制跨度（span of control）指的是受经理直接监督的雇员数量。一般说来，合理的控制跨度在 10 人左右。工人远远多于这个数字的产线将可能需要中等层级的管理（领班、首席技师、多层次的产线经理）。当然了，10 也仅仅是一条粗略的经验法则，合适的受经理直接监督的雇员数量由于部门而不相同。严格说来，考虑到经理要监督的产品或制程范围，控制跨度这个术语不应当单指下属的数量。

例如，印刷电路板（PCB）的制造包括，将铜和玻璃纤维压到一起的叠片工艺，蚀刻铜片获取所需电路的印制工艺，等等。上述两种工艺的技术、设备和物流都很不相同。叠片是包含大型机械挤压的成批处理，而印制则是一次一片使用光学曝光设备和自动化转运、使用化学蚀刻的加工组合。这些差别，再加上物理上的分隔，使得逻辑上应当为两种制程指派不同的经理。

产线如何分解，从瓶颈设计、控制跨度或其他方面考虑，都与 SFC 模块的结构相关。取决于产线的复杂性，经理们或许可以在从生产控制系统受助不多的条件下，协调通过他们

⁴ 注意到故意增加产能的想法会导致某些资源利用不充分，从而违背产线平衡的准则。对产线不平衡做出经济评判需要采取考虑到变动性的全线的眼光，正如我们在本书中一直强调的。

负责的产线的物料流动。但在管辖范围之外，他们就无能为力了。一般认为，高层的经理的职责比各部分的大（并且，理所当然地还有工厂经理对于整个工厂有着无限的责任）。可是，在实时控制的层级，这些高层经理无法通过强制来实现协调。它必须由产线经理依据 SFC 模块提供的信息来做到。

至少地，SFC 模块必须告诉经理们下游工站需要什么工件。如果 SFC 模块还能预报各个工站的来料，那就更好了，因为这条信息能使产线经理们提前计划他们的活动。(460|461) 产线处于管理目的的分划提供了产线位点的自然组合来报告这条信息。产线如何分划也会影响图 14.1 列出的 SFC 模块的功能。出于明确职责的考虑，应当在受不同管理的工站之间建立质量检查点（如，下游工站检查来自上游工站的工件，如果没有达到规格即拒绝接收）。在这些条件下，必须建立起 SFC 与质量控制之间的连接。

14.3 CONWIP 结构

如我们在第五章所见，JIT 作者们有时被花言巧语的简洁性欺骗，做出类似“使用一些容器和标签，看板……可以在……15 分钟内安置”（Schonberger 1990, p.308）。任何曾安置拉式体系的经理会知道，是一个系统运行良好并不简单或容易。制造企业有着复杂、多样的活动。“浪漫主义 JIT”的高级哲学方针或“实用主义 JIT”的技术组合，都不能为单个的制造环境提供现成的方案。记住了这一点，我们开始对可能的 SFC 结构的评论。我们始于最简单的可能性，发现它们在哪些地方运行不良，并移向适应于复杂环境的更精巧的方法。由于不可能详尽讨论每个主题，我们希望这里的范围能为读者提供具体应用所需选择和开发 SFC 模块的混搭（mix-and-match）起始点。

14.3.1 基础 CONWIP

从管理的立场来看，最简单的制造环境是单路线、单产品族的产线。如果下面的条件成立，则依据第十章的讨论，这个模型就是对现实的精确估计，并且基础 CONWIP 体系（其中投料由完成来调整，以保持产线中 WIP 为定值）将在 SFC 模块中运行良好：

1. 路线恒定故所有的工件经过相同的机器序列。事实上，如果某些工件包含并不显著影响流动时间的几项额外作业（如，安装豪华装饰物），我们就可以忽视它们并仍然使用基础 CONWIP。然而，如果路线是条件性的（如，加工任务可能送入重工线或代工商），我们就不能把产线当作单路线，需要的处理方法就比基础 CONWIP 多。

2. 加工时间差不多故所有的工件在各个加工中心处所需的时间近似相等。这意味着瓶颈将是稳定的。然而，我们并不需要清晰地识别（即，显著慢于其他机器）瓶颈。

3. 没有显著的换模时间故单个加工任务通过产线的时间不会强烈地受加工任务序列的影响。

4. 没有组装线故加工任务的进行可视为直线流动。我们稍后再修改基础 CONWIP 来适应组装线。(461|462)

维持常量 WIP 协议最简单的方法可能是通过物理的卡片或容器，如图 14.4 所示。原材料由标准容器送达到，但仅在有可用的 CONWIP 卡片时投入产线。卡片可以是硬纸板、金属或塑料标签，或者就是空容器本身。卡片上不需要路线或产品信息，所以它们可以非常简易。假如作业只依据卡片投入产线，卡片如实重复利用（它们没有因加工任务移向重工而停顿或因工程变更要求而结束），产线中的 WIP 将保持恒定于 CONWIP 卡片数目设定的水平。

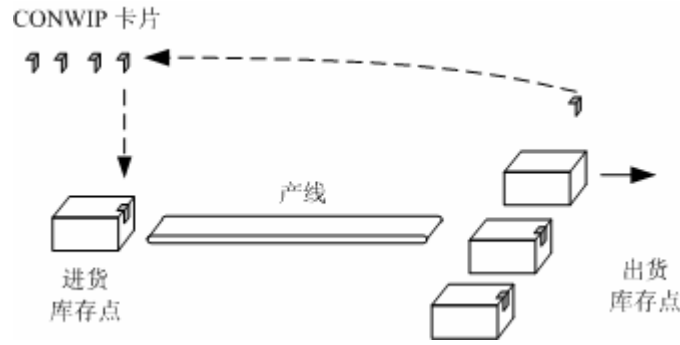


图 14.4 使用卡片的 CONWIP 产线

即使在这个简易系统中，也有要解决的 SFC 问题。

1. 作业延迟。由于 CONWIP 卡片并不包含产品信息，产线经理或作业员需要另外的信息来选择投入产线的加工任务。这是排序与排程（sequencing and scheduling）模块的任务，可能会用到简单的最早交期（EDD）排序法则（由于无换模时间的假定）或复杂的成批处理程序（在长期内加工类似工件以实现某个节奏）。一旦形成，作业延迟可以通过多种方式与产线沟通。最简单的是一张作业优先次序清单。一旦出现可用的 CONWIP 卡片，有原材料供应的下一个加工任务就投入产线。有些情形可能需要更精巧的作业延迟显示方法，例如，显示优先度或预期到达时间。

2. 产线准则。一般地，产线应当保持先进先出。它意味着，除了产出损失、重工或经过多机工站的情形，加工任务将按照它们的投入顺序离开产线。由于 CONWIP 协议保持产线以平稳的节奏运行，这样很容易就能预测加工任务——甚至包括那些处于作业延迟的——何时完成。然而，如果 CONWIP 产线很长，特定加工任务要加速的情形就多起来了。尽管我们因为剧烈提高了产线的变动性而贬低对加速的随意使用，期望企业永远不会使用加速也是不合理的。为了将结果的扰动最小化，很有必要只允许两种水平的优先级并且建立特殊的**超越点（passing points）**。超越点是产线中的缓冲或库存点，常常位于 CONWIP 环段之间，在那里“加急”加工任务可以超越“普通”的加工任务。工站从这样的缓冲取得原材料的准则是，如果有**加急清单（hot list）**，就取出那上面的第一个加工任务；如果没有，就取出当前该缓冲中历史最久的。只允许在产线中指定的位置处超越，使得建模（实时仿真模型）预测加工任务何时离开产线变得容易。如果设定多个优先级并且允许无限制的超越，产线中的变动性或“扰动”（churn）可能会变得非常剧烈，并使预测产线行为变得不可能。（462|463）

3. 卡片数量。出于有效性的考虑，CONWIP 产线的 SFC 模块必须设定一个合理的 WIP 水平。如我们在第十三章所见，设定卡片数量不应当频繁进行（如，月度或季度），不是与作业投放实时发生。如果在已有的产线上实施 CONWIP，较简单的方法是将卡片数量的初始值固定在其 WIP 的历史水平。待其稳定后，观察工站前的持久队列。如果某一工站处的队列事实上永远没有清空过，则削减卡片数量不会对产出较大影响所以应当实施。定期检查队列长度，从而调整卡片数量来适应产线的物理变化（有希望是改进）。如果在全新的产线上实施 CONWIP，合理的方法是通过选择合理、可行（feasible）的周期时间 CT 以及估计经验生产速率 r_b^P ，从而计算得到 WIP。依据里特定律，设定 WIP 水平为

$$WIP = CT \times r_b^P$$

假设实际产出接近于 r_b^P ，再如果 CT 真正可行，这种方法将得到合理的 WIP 水平。一定要小心不能过于乐观地估计周期时间，否则将低估所需的 WIP，从而导致产出下降。

4. 卡片不足。如果卡片数量相对于产线变动性来说足够大，严格坚持 CONWIP 协议能执行良好。然而，存在着我们被迫违反 CONWIP 投料准则的情形。图 14.5 显示了一种这样的情形，其中瓶颈之后的一台机器正经历一次长得离奇的失效，导致瓶颈缺乏卡片而饥饿。如果非瓶颈机器比瓶颈快得多，则它修好后会快速赶上。但在同时，我们失去了瓶颈处的宝贵时间。这种情形的一种补救措施是执行**卡片不足 (card deficit)**行动，向产线投入一些没有 CONWIP 卡片的加工任务。这将使得瓶颈重新开始工作。一旦失效的问题解决，我们回到 CONWIP 准则，并只允许有卡片地投料。不带卡片的加工任务将逐渐清出产线，WIP 将回落到目标水平。这类问题的另一种补救措施是在瓶颈而不是线尾拉，细节在 14.4.2 节讨论。

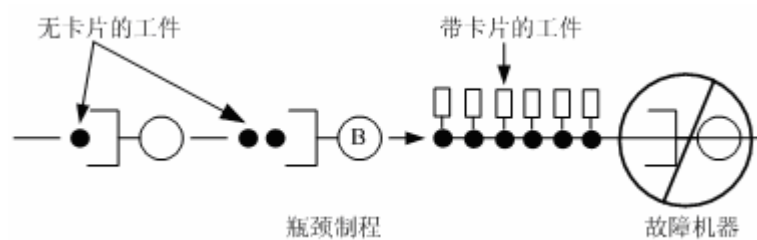


图 14.5 失效情形下 CONWIP 卡片不足

5. 提前作业。我们在第十章提出的 CONWIP 的一种好处是，它能在情况许可时择机先于计划作业。例如，如果本周内瓶颈异乎寻常地快或可靠，我们就可能比计划做得多。假设主生产计划排满了，有必要利用我们的好运气——做到上限。尽管启动下周的某些加工任务完全可行，启动数月后才交付的加工任务就不合适。如果某条路线的 MPS 没有排满，这对于许多路线只被零星使用的工厂也是常事，则我们会希望建立一个**提前作业窗口 (work-ahead window)**。

例如，通过 CONWIP 机制授权时，假设下一个加工任务距其交期还有 n 周 (*provided that it is within n weeks of its due date*)，我们将其投入产线。(463|464) 确定这个限制的 n 也是 CONWIP 设计中的一个附加问题，与第三章中讨论的冻结期 (frozen zones)、时界 (time fences) 等概念密切相关。处于交期的冻结期之内的加工任务不再对变动做出反应，所以就可以允许 CONWIP 对其先行作业。处于冻结期之外 (或部分限定时界) 的加工任务若先行作业就有较大的风险，因为客户需求可能会变动。显然地，合适的先行作业策略的选择强烈依赖于制造环境。

14.3.2 串联 CONWIP 产线

即使满足了基础 CONWIP 实施所需的条件 (恒定路线、差不多的加工时间、无明显换模时间以及无组装线)，我们可能也不会希望按单条 CONWIP 环路来运行这条产线。原因在于，控制跨度会促使我们将产线分化为易于管理的部分。一种实现方法是，通过 WIP 缓冲隔离出几个串联 CONWIP 环路加以控制。各个环路的 WIP 水平保持在各自的恒定值。环路之间的缓冲保持着足够的 WIP，使得环路能以各自的速度运行而不互相影响 (阻塞或饥饿)。这样会易于不同的经理管理不同的环路。这些缓冲带来的额外 WIP 和周期时间降低了效率，却是考虑到具体制造环境需求时不得不做出的权衡。

图 14.6 显示了单条产线的不同 CONWIP 分解方式，从将整条产线视为一个 CONWIP

环路到将各个工站视为一个 CONWIP 环路。注意到最后一种情形，即每个工站作为一个环路，等同于单看板系统。在某种意义上，基础 CONWIP 和看板是基于 CONWIP 的 SFC 结构的连续统。分解出的 CONWIP 环路越多，它的行为就越接近看板。如我们在第十章的讨论，看板提供了对各工站之间物流的更紧密控制，并且在 WIP 水平足够低的时候能促进邻近工站之间的通讯。可是，看板系统中要设定的 WIP 水平很多，在实施时就趋于比基础 CONWIP 复杂。故而，在决定划分多少 CONWIP 环路来控制产线时，除了考虑效率/控制跨度的权衡，还要考虑复杂性/通讯的权衡。

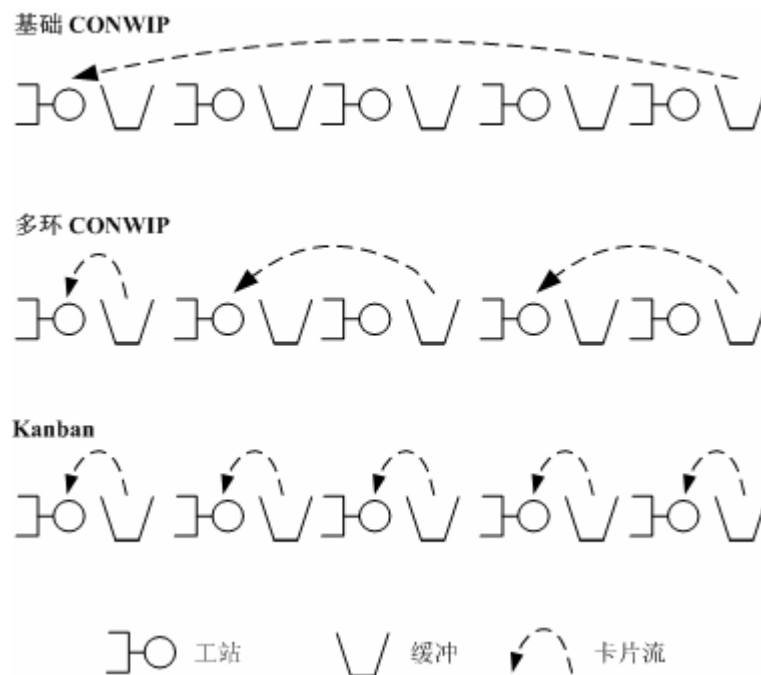


图 14.6 串联 CONWIP 环路

使用多个 CONWIP 环路控制的产线中，另一个问题是何时释放卡片。两个选项是，(1) 当加工任务进入环间的缓冲，(2) 当加工任务离开环间的缓冲。如果 CONWIP 卡片在环路尾端仍然系于加工任务，则产线 WIP 与缓冲 WIP 之和将保持恒定。故而，如果缓冲 WIP 达到卡片数目的上限，则该环路将在下游环路从缓冲中运走 WIP 和释放卡片之前保持关闭。如图 14.7 所示（环路 1 和 3），这种机制对于速度快到能与整个产线保持同步的非瓶颈环路很有意义。如果不通过在缓冲将加工任务与卡片分离的方法，使环路 1 连接到产线的节奏，它就可能远远快于其他环路，导致系统充满 WIP。

如果某一环路明确定义为瓶颈，我们就希望将它从产线中分离出来，从而使它尽可能快地运行（即，先行作业）。如图 14.7 中的环路 2 所示，我们的实现方法是一旦加工任务离开线尾——在它们进入下游缓冲之前——就释放卡片。这就可以在上游缓冲处 WIP 可得性与产线各点处 WIP 上限给定的情况下，使该环路的速度尽可能地快。(464|465) 当然了，这就意味着下游缓冲中的 WIP 可以无界漂移，但只有后续产线都比瓶颈环路快，它们就可以赶上来并因此使得 WIP 不会变得过大。同样确定的还有，从长期来看，所有的 CONWIP 环路都以同样的速率运行，都等于瓶颈速率。

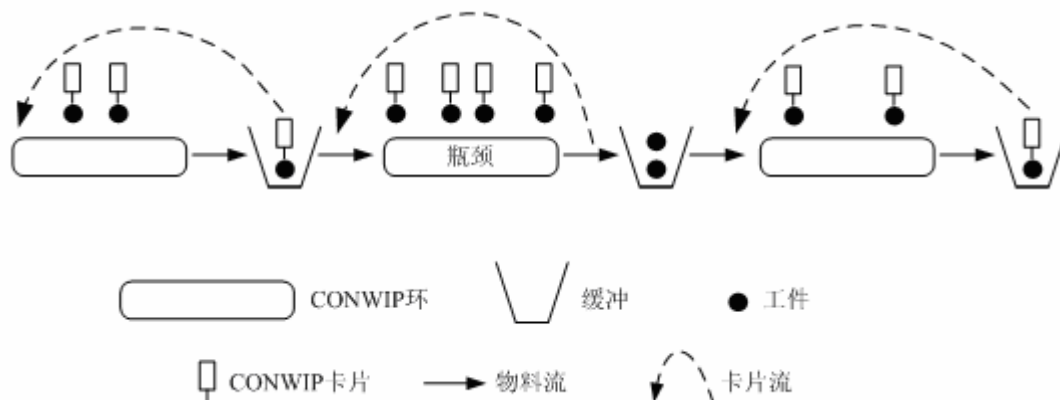


图 14.7 连接的与分离的 CONWIP 环路

14.3.3 共享的资源

从物流角度看, 各台机器指定到各个路线——有时正是指定产品族到制造单元的情形——显然是最简单的了, 但其他的一些考虑因素常常使其不可能实现。例如, 某台非常昂贵的机器被两种不同的产品所需要, 但路线的其他部分都不同, 这时购买两台同样的机器来分离两条路线就很不经济。其结果在某种程度上就像先前的图 14.3 的情形。如果几种多重资源被许多路线共享, 情形就变得相当复杂了。

共享的资源使 CONWIP 产线的控制和预测变得复杂。控制变得复杂, 是因为我们必须在多条进入的路线中选择加工任务执行作业。如果共享的资源在 CONWIP 环路内部, 则做出这个决定的自然信息是进入的加工任务的“等待时间”。(465|466) 合适的选择准则是先入系统先出 (FISFO), 原因是拉式系统中加工任务进入产线的时间对应于下游需求的时间。故而 FISFO 可以将生产和需求协调起来。

若要确保共享的资源对下游迫切需要的加工任务执行作业, 则有必要将共享的资源之前与之后的部分分为独立的 CONWIP 环路, 如图 14.8 所示。这幅图绘出了对应于产品族 A、B 的两条路线, 它们有一个公共资源。这样就在公共资源处产生了从上游缓冲送达的部件, 以及指示下游补足需求的卡片。先对等待时间最长的加工任务执行作业 (假定在进入的缓冲中有适当的物料), 是一种强迫共享资源加工后续最可能用到的工件的方法。如果产品族之间的切换需要机器换模, 则各族在切换之前应当加工多少件还需要附加的准则。

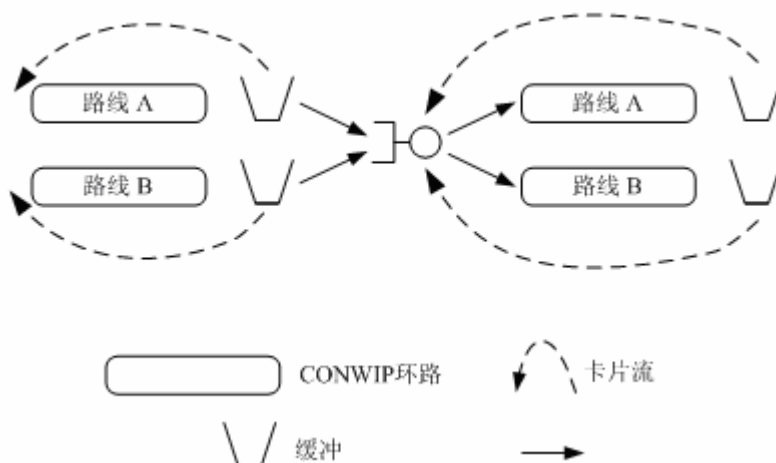


图 14.8 在共享的资源处分划 CONWIP 环路

共享资源也增加了预测的复杂度。传送带模型可以相当精确地估计加工任务离开单个 CONWIP 产线的时间，但与其他产线共享资源后就几乎没有精确性可言了。原因在于，一条产线的可能会输出强烈依赖于其他产线的情况。一种调整传送带模型来适应这种情形的简单方法是，预先分派产能。例如，假定对应于产品组 A、B 的两条 CONWIP 产线分享一处公共资源，其中族 A 利用 60% 的时间，族 B 利用 40% 的时间。然后我们可以将族 A 在共享资源的加工时间除以 0.6 来表示该资源仅投入 60% 的时间给族 A。类似地，我们将族 B 在共享资源处的加工时间除以 0.4 来对族 B 的产线进行处理。

为了稍微详细一些地说明上述分析，假定图 14.8 所示的共享资源需要路线 A 的加工任务每小时一件，需要路线 B 的每两小时一件。如果此处 60% 的加工任务来自路线 A，40% 来自路线 B，则投入到 A 的加工时间（用于生产的时间）的比例为

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

因此，投入到 B 的加工时间是 $1 - 0.4286 = 0.5714$ 。42.86% 这个数字很像机器断供引起的可用率 (*availability*)。(466|467) 它等效于，该资源对 A 仅仅有 42.86% 的时间是可用的。故而，当共享资源处仅有 A 的工件作业时，速率为 1 件/小时；但由于与 B 分享，其速率降到 1×0.4286 件/小时。平均加工时间是这个速率的倒数， $1 / 0.4286 = 2.33$ 小时/件。类似地，加工一件 B 的平均时间为

$$\frac{2}{0.5714} = 3.50 \text{ 小时/件}$$

在共享资源处采用这些扩大的加工时间，就可以在分析时将路线 A、B 视为完全分离的 CONWIP 产线。当然，如果两条路线的容量剧烈波动，加工任务输出时间就会在传送带模型预测值的上下显著变化。其效果很像 CONWIP 产线中某处资源有着高度变动（如，长期不频繁，与短期频繁相反）的断供时间。故而，若使用这样的模型来设定交期，我们将不得不附加一个较大的宽放系数来补偿额外的变动性。

14.3.4 多产品族

我们现在通过考虑产线有多产品族的情形，开始松弛基础 CONWIP 成立所需的假设条件。我们仍然假设对象是一条简单流水线，无装配线，但现在允许有显著差异加工时间以及换模时间可能依赖于序列的不同产品族。在这些条件下，通过保持产线中的工件恒定来固定 CONWIP 环路中的 WIP 水平就不再合理了。原因在于，产线的总负载可能由于不同产品加工时间之间的差异而变化巨大。这时候更需要调整 WIP 来适应产能。

一种看似合理的 WIP 量度是瓶颈机器的加工时间。这种方法认为，如果在瓶颈处产品 A 每件需要一小时而产品 B 需要两小时，则每当一件 B 离开产线时我们允许两件 A 进入（假设它在作业积压单上是下一个）。只要瓶颈位置对产品组合相对不敏感，这种机制就趋于保持瓶颈的负载稳定。如果瓶颈随产品组合漂移（即，不同产品的最慢资源不同），则计算一个按产能调整 (*capacity-adjusted*) 的 WIP 水平就较为复杂。我们可以使用所有机器处的总加工时间。然而，这时需要的 WIP 水平比瓶颈稳定的系统更高，以补偿引起瓶颈漂移的变动性。还有，如果不同产品的加工时间区别不大，这种方法与 WIP 的简单物理计数也没什么不同。

如果 WIP 的计数单位是按产能调整的标准件，用卡片这样的简单机制控制 WIP 水平就变得更困难了。与其努力用多种卡片来反映加工任务的不同复杂度，还不如使用电子系统来监控 WIP 水平。图 14.9 显示了一种电子 CONWIP 控制器 (CONWIP controller)，它是一个由位于线首和线尾的计算机组成的局域网 (LAN)。计算机监控调整后的 WIP 水平，并在

它落到目标水平以下时作出报告（如，通过将指示灯的颜色由红变绿）。发生此种情况时，第一个工站的工人选出作业积压单上所需物料齐备的下一个加工任务（交期、料号、数量等信息显示于计算机终端），并投放到产线。（467|468）这次投料由键盘或光学扫描仪记录下来，并添入按产能调整的 WIP 水平。在产线的末尾，加工任务的离开也被记录下来并从 WIP 水平中减去。例外事故，例如产出损失的附带结果，也需要在某个计算机终端被记录。

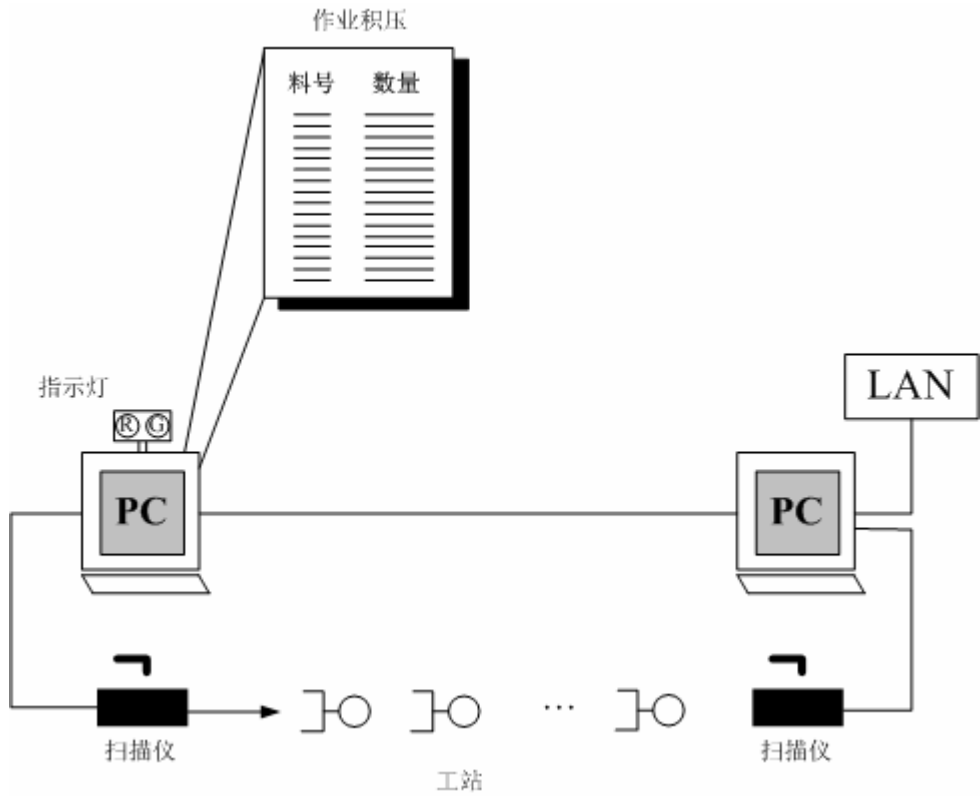


图 14.9 使用电子信号显示的 CONWIP 产线

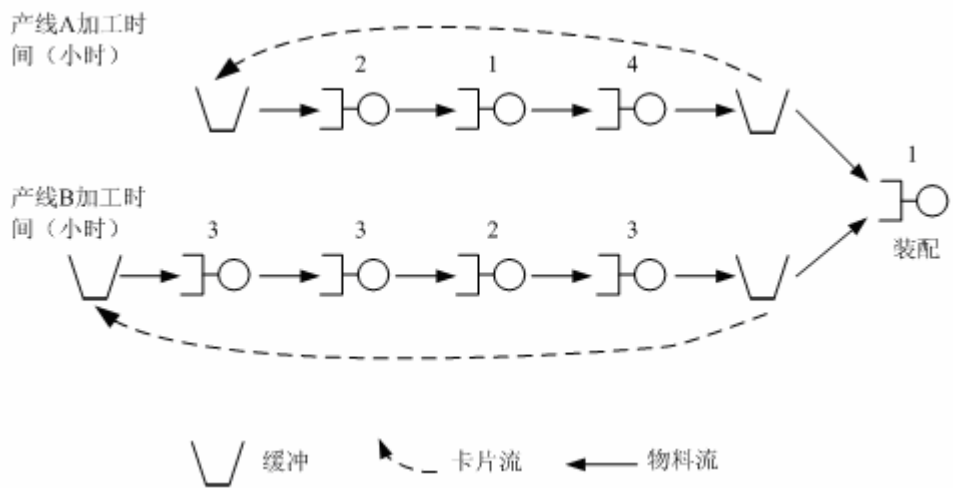


图 14.10 装配制程的 CONWIP 控制

14.3.5 CONWIP 装配线

现在我们将 CONWIP 概念扩展到有装配作业的系统。图 14.10 展现了两条制造产线供给一道装配作业的简单情形。每次装配需要来自族 A 的一个组件和来自族 B 的一个组件。两个组件齐备，装配作业才能开始。两条制造产线都由 WIP 水平固定，却不一定相同的 CONWIP 环路控制。每当装配作业完成，一道信号（如，CONWIP 卡片或电子信号）就触发每条产线进行一次新的投料。只要产线维持了 FIFO 协议，最终装配序列就与投料序列一致。

注意到装配的完成不必触发投放出预定给同一次装配的组件（Notice that assembly completions need not trigger releases of subcomponents destined for the same assembly）。如果产线 A 的 WIP 水平为 9 件，产线 B 的 WIP 水平为 18 件，则下一次装配完成授权的投料对于产线 A 将在从现在开始 9 次装配之后用到，产线 B 将在 18 次之后用到。（468|469）若产线 B 的总加工时间比 A 长，这种不平衡是可行的。一般来说，较长的产线需要较大的 WIP（即，由于里特定律）。决定 WIP 的确切水平需要技巧。幸运的是，只要产线有足够多的 WIP 能防止瓶颈多次饥饿，绩效在这个 WIP 水平下将是稳健的。

为了说明在装配系统中如何设定合适的 WIP 水平，考虑图 14.10 给出的数据。注意到整个系统的瓶颈是产线 A 的机器 3。故而，瓶颈速率是 $r_b = 0.25$ 件/小时。如果我们将连带装配的两条制造产线视为单独的，则可以对它们使用第七章中的临界 WIP 公式。理想（即，完全确定）情形下的 WIP 水平需达到

$$W_0^A = r_b T_0^A = \frac{1}{4}(2+1+4+1) = \frac{8}{4} = 2$$

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

来实现全部产出。当然，实际的产线中存在变动性，所以 WIP 水平需要比它们大。具体大多少，取决于产线中的变动性有多大。

针对第七章中讨论的实际最差情形（practical worst case）产线，我们可以通过设定产出表达式等于 $0.9r_b$ ，计算得到出产出等于瓶颈速率 90%所需的 WIP 水平 ω ：

$$\frac{\omega}{W_0 + \omega - 1} r_b = 0.9r_b$$

$$\frac{\omega}{W_0 + \omega - 1} = 0.9$$

$$\omega = 0.9(W_0 + \omega - 1)$$

$$\omega = 9W_0 - 9 = 9(W_0 - 1)$$

将 W_0^A 、 W_0^B 带入上式得

$$\omega^A = 9(2-1) = 9$$

$$\omega^B = 9(3-1) = 18$$

除非产线是高度变动的，否则这两个 WIP 水平可以作为一个合理的起点，之后再进行不断

的调整过程。如果所有机器的加工时间都比实际最差情形的变动性小（即，它们的变异系数小于一），则产线能以小于上述值的 WIP 水平有效运行。如果所有机器的加工时间都比实际最差情形的变动性大（即，由于长时间的失效或换模等原因，变异系数大于一），则需要比上述值更大的 WIP 水平来实现一个合理的产出速率。

14.4 其他拉式机制

我们将 CONWIP 作为所考虑的 SFC 平台的首选。它是简易的、可预测的，还是稳健的。因此，除非制造环境非常特殊使得它不适用，或者其他方法能产生明显更好的绩效，CONWIP 总是一个良好、安全的选择。通过使用上述讨论的将物理产线分拆为多个 CONWIP 环路的这种柔性，我们可以满足各种广泛的环境对 CONWIP 的需求。可是也有一些拉式情形中的 SFC 模块，不能命名为 CONWIP。下面我们讨论几种可能性。(469|470)

14.4.1 看板

正如我们先前提到的，看板可视为串联 CONWIP 环路的一种极端形式——每个环路中仅有一台机器。所以在 CONWIP 热心者看来，看板只是 CONWIP 的一种特殊情形。然而，大野书中看板系统的图画很想一系列 CONWIP 环路供给一条装配线。因此，JIT 的开发者已把 CONWIP 当作看板的一种形式。在我们关心的范围内，这种区别仅仅是语义学上的；看板和 CONWIP 显然有着密切的联系。重要的问题是，何时使用看板（单工站环路）代替 CONWIP（多工站环路）。

看板比 CONWIP 多了两种潜在优势：

1. 通过促成各工站都从上游工站处拉，看板可能迫使工站之间有更好的通讯。尽管也有其他促进通讯的方法，看板使这个过程几乎自动化。
2. 通过在每个工站处分划产线，看板自然地提供了如图 14.8 所示的机制，在不同的路线之间分享资源。

然而，看板也有如下的潜在劣势：

1. 它比 CONWIP 复杂，需要更多的 WIP 水平的规格。（然而，回忆起拉式系统对 WIP 水平相当不敏感，故看板的 WIP 水平不需要为系统运行良好而精确设定，所以这项复杂性的提升对大多数看板的实施不是主要障碍。）
2. 它更严格地控制产线节奏，降低了工人先行作业的柔性，对工人迅速补充缓冲施加了相当大的压力。
3. 按产品分类的（product-specific）卡片意味着各工站处都至少有各种料号的一个标准容器，使得下游工站能拉到它们需要的。这对于料号众多的系统来说不切实际。
4. 它不能适应变化的产品组合（除非将大量的 WIP 送入系统），原因在于按产品分类的卡片数量严格管制系统中的 WIP 组合。
5. 它对少量、不频繁的订单（一件或两件）来说不切实际。要么 WIP 不得不长时间（即，两次订单期间）停滞无用，要么系统不响应这类订单——看板卡片的授权信号必须由后向前经过整条产线传递到线首来触发新 WIP 的投放。

前两种劣势很难减轻；复杂度和压力是看板为额外的局部控制所付出的代价。然而，其

余的劣势产生于按产品分类的卡片并因此可以通过使用**按路线分类的卡片（routing-specific card**，或称路线有别的卡片）或**作业积压单（work backlog）**来消除。图 14.11 展示不同颜色的卡片对应不同路线的看板系统。（470|471）当标准容器从出货库存点（outbound stock point）移出时，卡片授权生产来替代它。将要生产的工件的编码由排序和排程模块建立的作业积压单决定。如果某一工件长期未出现在积压单上，它就不会出现在产线中。改用按路线分类的（对应于按产品分类的）卡片使看板课按这种方法用于多料号系统。

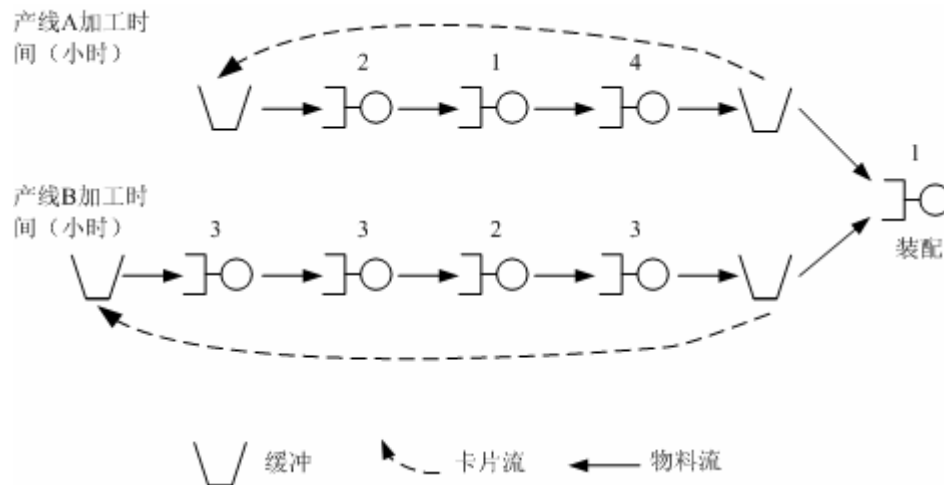


图 14.11 有按路线分类的卡片和作业积压单的看板

通过以上讨论的可以看出，看板似乎最适合于有着许多路线和共享资源的系统，特别是那些产品和路线频繁增加或移除的。如果我们要将产线分划为许多 CONWIP 环路来更简易地控制共享资源，则一直改到看板并不会显著地改变绩效。还有，如果一条新的路线由明显的非共享资源改到共享资源，则看板这种布局就已经为之提供了所需的分划。

另一方面，如果各种路线少有共享资源并且新产品趋于遵循已有的路线，这是就没有什么动力去引入看板的复杂性。这种系统很可能在 CONWIP 下更简单和有效地运行，此时的产线出于控制跨度的原因可能要分划独立的环路，对共享资源给予特别关注，或者在装配点处补给缓冲。

14.4.2 从瓶颈处拉

CONWIP（或看板）在特定环境下可能引起的问题有：

1. 下游机器失效引起的瓶颈饥饿（*bottleneck starvation*）。如图 14.5 所示，我们希望允许非卡片授权的投料来补偿这种情形。

2. WIP 水平保持恒定的需求引起的过早投料（*premature release*）。即使某个工件在几个月内都不需要，由于环路中的 WIP 降到目标水平之下，CONWIP 系统可能还是会启动投料。这会因为一些不好的原因而降低柔性（如，一旦加工任务投放，将很难对工程变更或客户需求变更做出相应调整）。（471|472）

我们可以通过修正 CONWIP 来解决这些问题。基本的想法是设计一种机制，使瓶颈能先行作业，但同时也防止提前地过多。我们将介绍的技术与高德拉德（Goldratt and Fox 1986）开发的名为**鼓-缓冲-绳子（drum-buffer-rope, DBR）**的技术相关，尽管高德拉德最初提出 DBR 是作为一种排程方法论而非 SFC 机制。

我们以最简单的**从瓶颈处拉**（pull-from-bottleneck, **PFB**）策略开始。图 14.12 显示了一条单产线形式的这种系统。它与 CONWIP 的不同之处在于，WIP 水平在瓶颈以及瓶颈的上游保持恒定，但在瓶颈之后随意浮动。由于瓶颈下游的机器平均来说比瓶颈快，WIP 通常不会在产线的这部分堆积。然而，即使这些机器中的失效引起 WIP 暂时堆积，也不会使瓶颈关闭，而在 CONWIP 下如果没有启动卡片不足措施瓶颈就会关闭。因此，PFB 保持了瓶颈的稳定，可以作为产线中卡片不足措施的一种替代方法。如果瓶颈随产品组合发生变化，则拉的位点难以确定，我们就不得不从产线的末尾拉动（即，使用常规的 CONWIP），可能还要使用卡片不足策略。

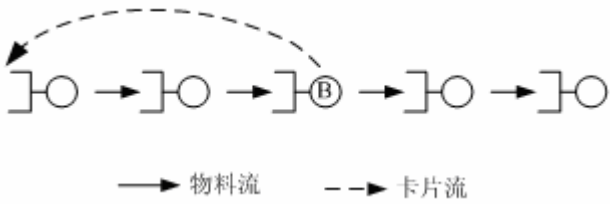


图 14.12 一个从瓶颈处拉系统

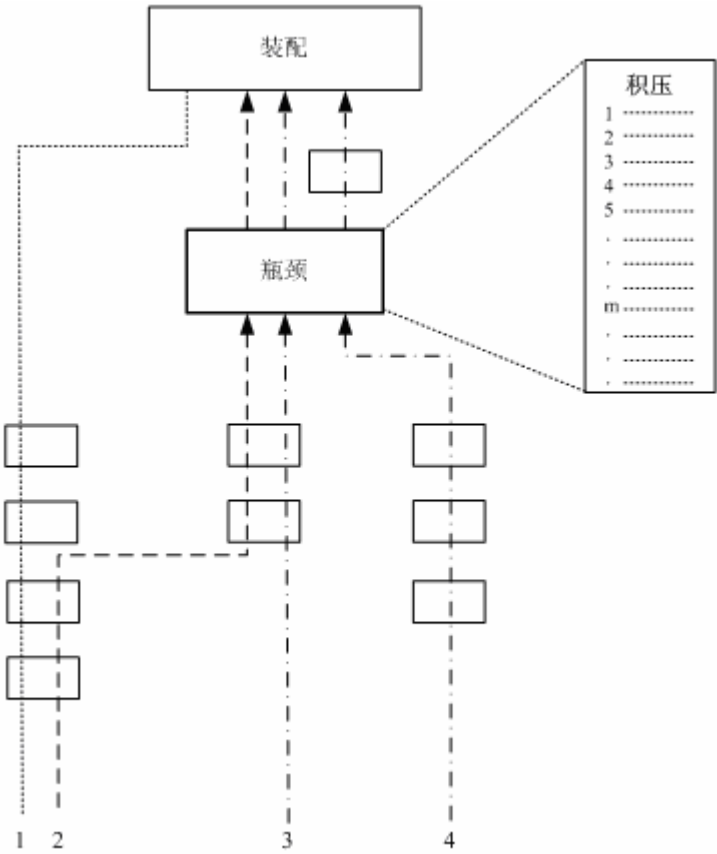


图 14.13 一个加工车间中的路线

图 14.12 所示的简单 PFB 方法可以减轻与 CONWIP 相关的瓶颈饥饿问题，却不能解决过早投料的问题。当谈及一条单产线时，我们常常认为产线的运行将维持在接近全部产能的状态。在仅有几条产线的工厂中，这常常是正确的；但在产线众多（如，接近于加工车间布局的工厂）的工厂中，一些路线并不会长时间使用。例如，我们见过有 5,000 条不同路线的

工厂，其中仅有相当少的几条在任何时候都保有 **WIP**。显然地，在这些情况下我们不想在路线中维持常量 **WIP** 水平，否则将释放出很久以后才用得到的加工任务。

考虑图 14.13 所示的情形，四条路线中的三条经过瓶颈。**PFB** 策略的目标是确保加工任务的释放，并在被需要之前到达瓶颈（即，等待的加工任务将在瓶颈之前形成缓冲，防止随机变异引起瓶颈饥饿）。

为了使这个方法精确，设定

b_i = 加工任务 i 所需的瓶颈加工时间。注意不同路线的加工任务可能有着不同的加工时间，同一条路线中的不同产品族也可能有着不同的加工时间。

l_i = 加工任务 i 释放之后到达瓶颈的平均时间。注意这个时间仅包括在非瓶颈资源处的加工。由于大多数的队列出现在瓶颈之前，对于给定的路线这个时间应当是相对恒定的。⁵ 但这些时间可能在不同的路线之间区别很大。

L = 加工任务在瓶颈之前的缓冲等待的特定时间。这是由使用者决定的常量，取决于瓶颈处需要多长的保护时间。

现在我们通过累加 b_i 来计算产线瓶颈处的作业量。假设作业积压单上加工任务的序列就是它们在瓶颈处的作业序列，并假设加工任务 1 代表当前在瓶颈处执行作业（ b_1 代表它的剩余加工时间），则距离瓶颈对加工任务 j 可用的时间是

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

我们的目标是释放积压单上的加工任务，使其恰好在瓶颈之前等待 L 单位的时间。由于平均需要 l_j 的时间抵达瓶颈，我们可以在

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

的时候释放加工任务 j 。（473|474）所以，若我们对每个加工任务计算

$$\sum_{i=1}^{j-1} b_i - l_j - L \quad (14.1)$$

并在这个值达到零（或开始低于零）时投料，则可以维持瓶颈处的恒定负载，并保证加工任务在被需要之前在瓶颈处等待 L 单位的时间。只要 L 长到能够防止瓶颈处的显著延迟，瓶颈处的实际作业序列就可以相当好地匹配作业积压单上的序列。

注意到如果不同路线的提前期 l_j 不同，则投料序列可能与作业积压单上的序列不同。其他条件相同的情况下， l_j 较大的加工任务比 l_j 较小的先投放，因为它的（14.1）式系数较快

⁵ 这与假设包括瓶颈在内的整个工厂拥有恒定提前期的 **MRP** 恰恰相反。原因在于 **MRP** 在工厂内不保有恒定的负载，实际周期时间可能会非常大，使得恒定提前期的假设苍白无力。

地变成负值。还有，由于作业积压单上存在某些路线上加工任务不被需要的时间间隔，这个系统可能会在某些时点使某些路线的 WIP 降到零。故而，这种机制有 WIP 上限，但从维持路线恒定负载的意义上说它不是 CONWIP。

以上描述的 PFB 逻辑对路线 2、3 和 4 都很恰当，却不适用于路线 1，因为它不经过瓶颈。⁶明智的方法是将其作为一条 CONWIP 环路来控制。只要对来自路线 1 的工件的需求相对稳定，这样做就有效。如果最终装配序列包含路线 1 的工件不被需要的时间间隔，则我们会希望修改 CONWIP 逻辑来表达工件在投放前的一定时间窗口（如，一周）之内被需求的讯息。这样，我们就可以在（1）路线 1 的 WIP 水平低于它的目标值，以及（2）下一个工件在特定的时间窗口之内被需要这两种情况下投料。

14.4.3 车间作业控制与排程

保持工件直到它们在交期的时间窗口之内，这一点揭示了车间作业控制模块与排序和排程模块之间的潜在强烈联系。如果已经由排序和排程模块生成了一个进度表，则可以在符合 WIP 上限要求的条件下，按照这个进度表来投料从而对单条路线实施控制。也就是，只要路线的（按产能调整的）WIP 水平低于目标值，或者加工任务在其排定的投放时刻的一定时间窗口之内，就投放这个加工任务。如果进度表中有保持路线满载的足够作业，这种方法就等同于 CONWIP。如果某些产线上工件的进度表中有间隙，则其中的 WIP 水平可能会降到目标值以下，甚至降为零。

许多排配体系可用于以这种方式连接 WIP 上限机制。我们将在第十五章中讨论能非常好地适应这个目的的基于传送带模型的排程方法。（474|475）可我们也能使用一些不那么理想的，如 MRP。MRP 生成的计划投入量就代表一个进度表。与无视工厂的运行情况就按照这些计划投料相反，我们可以对 WIP 水平过高的路线阻塞投料，而对 WIP 水平过低的路线加速投料（到一个特定的数量）。MRP 的固定提前期假定仍趋于使进度表不准确。通过强制符合 WIP 上限，这种 SFC 方法至少会防止可怕的 WIP 爆炸。MRP 体系中设置 WIP 上限的好处很早就 MRP 文献（Wight 1970）中被提出，但生产实践中达到这种效果的机制却很鲜见。

14.5 生产追踪

如我们论及的，SFC 模块是工厂实时演变的重点（the SFC module is the point of contact with the real-time evolution of the plant）。因此，它成为监控工厂行为的自然位置。我们对关注制定进度表的短期，以及出于计划目的收集准确数据的长期都很感兴趣。尽管单间工厂可能对数据有着广泛的特殊需求，我们将注意力集中于两种一般主题：在短期监控朝向满足进度表的过程，在长期追踪用于其他计划模块的关键产能参数。

14.5.1 统计产出控制

在短期，基本的问题是我们是否能实现计划。如果产线按 CONWIP 环路运行且有特定的生产定额，那么问题就变成我们能否在期末（如，在一天或一周的末尾）完成定额。如果我们遵照路线的进度表，那么问题就是能否按时达到下一个加班点。若有落后于进度表的风险，我们可能会希望准备加班（先通报工人们）。若 SFC 模块能提供当前严重落后于进度的

⁶ 观察可知路线 2 与路线 3 共享非瓶颈资源，但我们并没有在投料机制上考虑这一点。只要这些共享的资源并不接近于成为瓶颈，不考虑也可以运行良好。然而，若这些资源由于产品组合而可能成为瓶颈，则需要更复杂的排程和投料方法了。细节讨论见第十五章。

警报，我们就可能重新分配资源或者采取其他矫正活动来进行补救。

我们可以使用类似于统计过程控制（SPC）中的技术来回答基本的短期生产追踪问题。由于很像 SPC，我们将 SFC 模块中的这种功能称为**统计产出控制（statistica throughput control, STC）**。我们考虑单周期内一个 CONWIP 环路的生产，来察看 STC 的运行机理。一般的周期有，（1）八小时的班次（加班时还可移用四小时的预防性维护时间），（2）第一班和第二班（第三班可用于加班），（3）周一到周五的规则时间（周六和周日可用于加班）。

标记期初为时刻 0，期末为时刻 R 。对任意有 $0 \leq t \leq R$ 的中间时刻 t ，我们必须比较两条信息：

$n_t = [0, t]$ 时间内产线的实际累积产量，可能要以按产能调整的单位

$S_t = [0, t]$ 时间内产线的计划累积产量

首先，注意由于 S_t 表示累积的计划产量，它往往随 t 而上升。然而，如果测度实际产量的点在可能发生产副作用的检查点之前， n_t 可能会下降。（475|476）其次，注意若产线使用了详细的进度表， S_t 可能会不均衡地上升。然而，若使用了周期性的生产定额而非进度表，则其目标是在时刻 R 之前完成 Q 件。假设 S_t 在各时段内线性（即，常量）增长，则

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

并有 $S_R = Q$ 。图 14.14 给出 S_t 的两种可能情形。

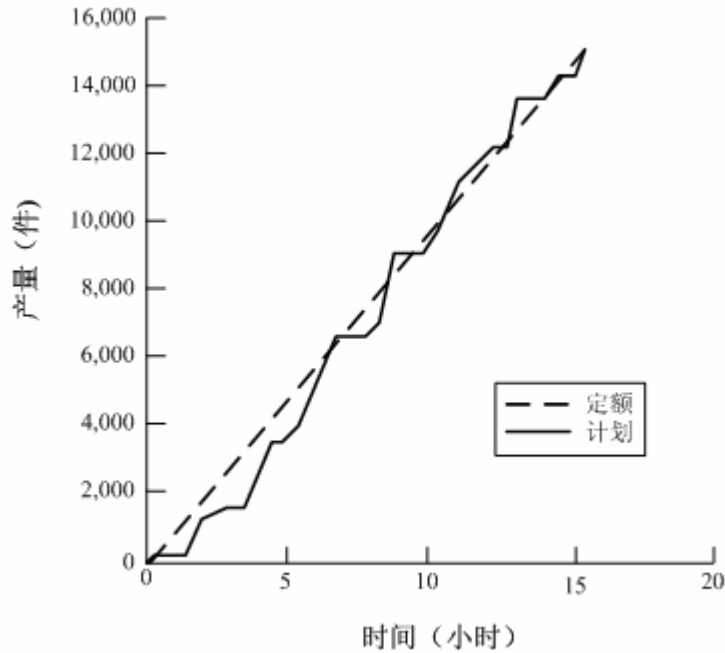


图 14.14 计划的累积产量函数， S_t

理想情况下，我们希望实际产量 n_t 在 $[0, t]$ 时间内处处等于计划产量 S_t 。当然了，由于工厂内的随机变动，它永远不会发生。因此，我们关注刻画现在超前或落后于进度表多久。可以绘制作为时间 t 函数的 $n_t - S_t$ 曲线，以产量来显示这种信息。当 >0 ，我们超前于进度；

当 $n_t - S_t < 0$ ，我们落后于进度。然而， n_t 与 S_t 的差值并未给出补足短缺的难度或过剩带来多大缓冲能力的直接信息。因此，更有意义的信息是在常规时间周期的末尾达到计划的可能性（*probability of being on schedule by the end of the regular time period*），它能指示现在超前或落后多久。

附录 14A 中，我们在可以使用正态分布估计任意时间段内产量分布的假设下，推导出上述可能性的一种表达式。从实际执行的角度看，很方便使用附录 14A 的公式来将引起错过定额的过剩水平（即， $n_t - S_t$ ）预设为一个特定的值 α 。如果知道了常规时间内产量（按产能调整的）的均值和标准差，标记为 μ 和 σ ，则定义 x

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

其中 z_α 在标准正态分布表中查阅 $\Phi(z_\alpha) = \alpha$ 得到。附录 14A 的意思是，若 t 时刻的过剩水平等于 x （即， $n_t - S_t = x$ ），则错过定额的概率恰等于 α 。若 $n_t - S_t > (<) = x$ ，则错过定额等概率就小于（大于） α 。

我们可以以简单的图形显示这些信息。图 14.15 绘制出特定的错过（未达到）定额概率对应的 x 值。（476|477）图中的曲线分别对应于 5%、25%、50%、75% 以及 95% 的概率。这个例子中我们假定有生产定额，其中常规时间有两班，共 16 小时，并且历史数据显示 16 小时的平均产量为 15,000 件， $\sigma = 2,000$ 件。定额设置为平均产能，即 $S_t = Q_t / R$ ，其中 $Q = \mu = 15,000$ 。图 14.15 的曲线给出如何设置定额的直观感觉。例如，若 t 时刻的过剩水平（即， $n_t - S_t$ ）正好落在 75% 曲线上，则达不到定额的概率就是 75%。有了这条信息，产线管理人员就可以采取措施（如，替换工人）来加速生产。若 $n_t - S_t$ 越过 50% 曲线，说明措施成功了。若 $n_t - S_t$ 下降了，比如说，在 $t = 12$ 时低过 95% 曲线，则达到定额越来越不可能，这时就要宣布发生超时了。

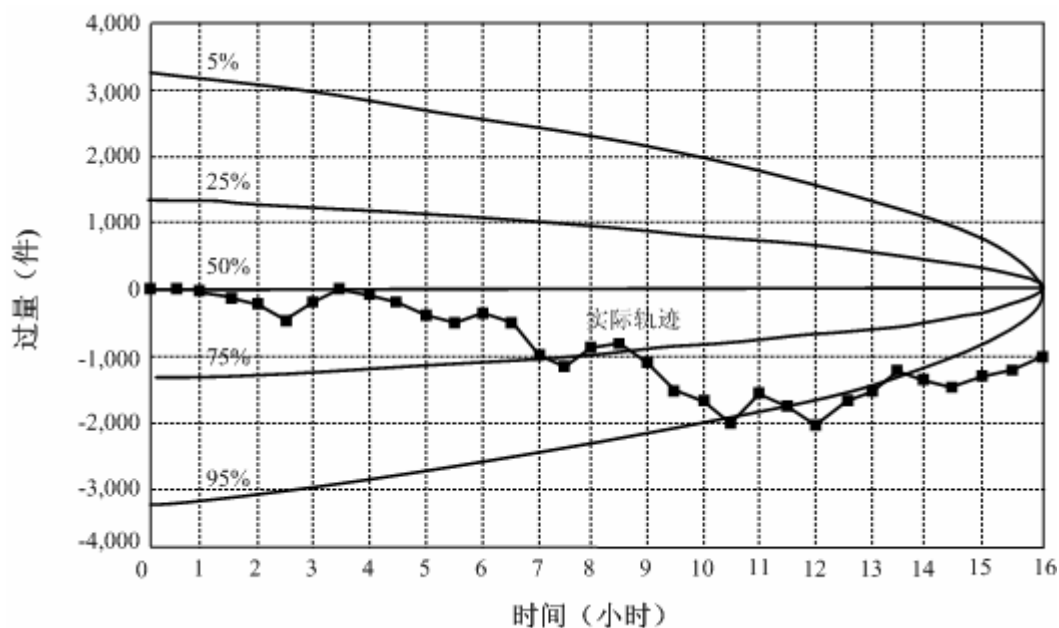


图 14.15 定额等于产能的 STC 图

注意到图 14.15 中对应于 $\alpha = 0.5$ 的值（即， x ）总是零。其原因是，由于定额严格设置为平均产量，当恰好准时时我们总有 50-50 的可能性来最终实现它。其他的重要值体现在曲线上。例如， $\alpha = 0.25$ 的曲线意味着在常规时间段的早期必须远远超前于计划的生产来保证完不成定额的可能性为 20%，但在常规时间段的末期超前一点就可以保证同样的可能性。原因当然是在时段的末期定额所剩无几，为实现它所需的过量生产也减少了。

第十三章关于拉式系统定额设置的讨论指出，将定额设在常规时间均值之下有着在经济上很划算。当它成立时，我们仍可以用式（14.2）来预先计算达不到定额的各种概率对应的过量值。图 14.16 图示了定额 $Q = 14,000$ 件的情形，它低于平均常规时间 $\mu = 15,000$ 件。注意到在这个例子中，如果我们始于无短缺或过剩（即， $n_t - S_t = 0$ ），则最初的达到定额的可能性大于 50%。这是因为我们已经将定额设置在平均常规时间的产量之下。由于 $Q < \mu$ ，平均来说我们应该可以实现一种节奏，使得 $n_t - S_t$ 为正值并持续增加。也就是说，定额达到时，或者生产停止，或者我们对下一期的定额先行作业。假如某处出了问题，不能超越节奏，这时若假定可以在 t 到期末的时间段内按历史平均节奏作业， $n_t - S_t$ 曲线使我们迅速识别达到定额的可能性。（477|478）

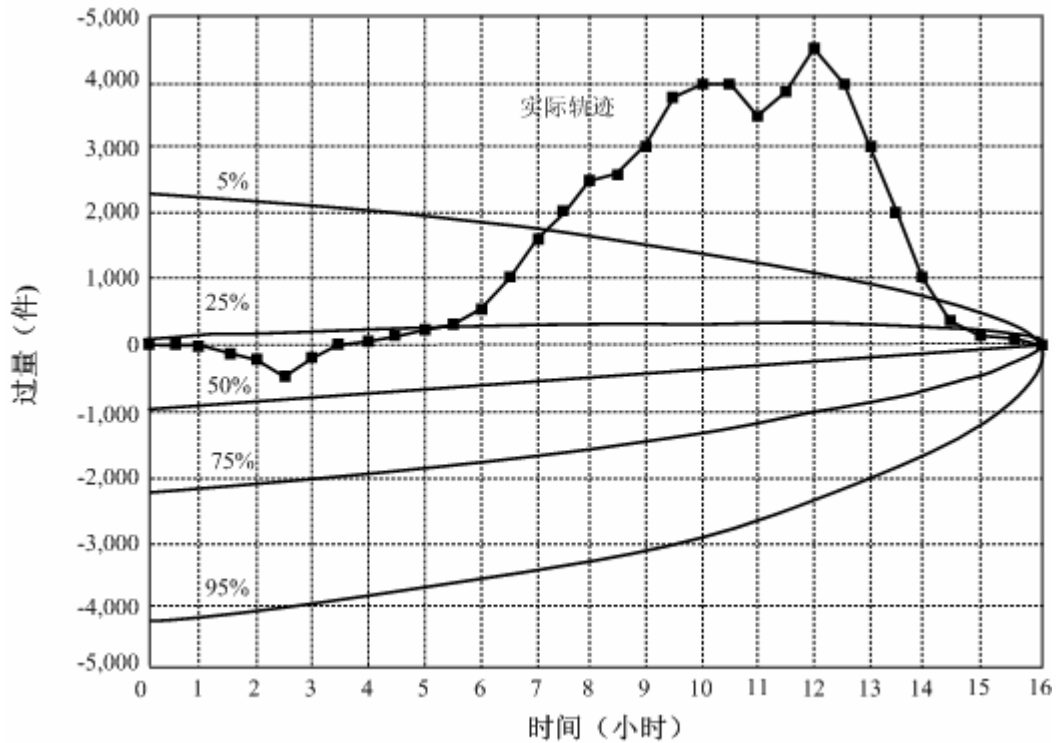


图 14.16 定额低于产能的 STC 图

图 14.15、14.16 之类的 STC 图可以根据式 (14.2) 和实际产量数据 (即, n_t) 绘出。

CONWIP 产线中, CONWIP 控制器的计算机终端 (见图 14.9) 是显示这些图的自然位置。STC 图也可以在工厂任何重要资源处保存和显示。

STC 图在 n_t 未被实时追踪时仍有用处。例如, 当常规时间是周一到周五, 我们只在每天的末尾获取实际产量的读数, 仍可以每天更新 STC 图来指示完成定额的可能性。

最后, STC 图对于多条路线共享的关键资源尤为有用。例如, 经过镀铜制程的两条电路板产线可以维持各自的 STC 图。产线管理人员根据两条路线的定额状态, 决定谁来使用镀铜制程。如果产线 1 安全地先于定额而产线 2 落后于定额, 则当有来件可用时应当使产线 2 执行作业。当然了, 我们应当明智地使用 STC 图, 在切换需要显著换模时避免产线之间的剧烈切换。

14.5.2 长期产能追踪

除了给工人和经理提供短期信息, 生产追踪系统还应当提供输入给其他计划职能, 如集结计划与劳动力计划以及定额设置等。这些职能所需的关键数据是以标准作业单位计的常规时间产量的均值和标准差。我们一直通过 SFC 模块来监控输出, 所以它是一个收集此项信息的合理位置。

在接下来的讨论中, 假设我们可以直接观察到常规时间内完成的作业数量 (如果可以的话, 使用按产能调整的标准单位)。对于严格定额系统 (只要定额完成, 即使发生在常规时间之前, 立即停止作业), 不应当使用这种程序, 因为它将低估实际常规时间产能。相反地, 应当收集完成定额的时间 (*time to make quota*) 的均值和标准差, 而这个时间可能比常规时间段长些或短些, 然后转化为常规时间产量的均值和标准差。(477|478) 实现这种转化的公式由 Spearman 等 (1989) 给出。

由于常规时间的实际产量会因随机扰动而上下波动，有必要平滑过去的的数据来估计得到对噪音不太敏感的产能参数。指数平滑技术（附录 13A）正好适合。我们可用这种方法来通过过去的输出观测值预测未来的产能。

令 μ 和 σ 分别表示常规时间产量的均值和标准差。这两个是我们期望从过去预测得到的值。令 Y_n 表示常规时间产量的第 n 个观测值， $\hat{\mu}(n)$ 表示常规时间产能的第 n 个平滑估计值， $\hat{T}(n)$ 表示第 n 个平滑趋势， α 和 β 是常数。迭代地计算 $\hat{\mu}(n)$ 和 $\hat{T}(n)$ 如

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

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

在每个常规时间段的末尾，我们接收一个输出 Y_n 的新观测值，再计算平均常规时间产能的估计值 $\hat{\mu}(n)$ 。为了开始迭代过程，还需要 $\hat{\mu}(0)$ 和 $\hat{T}(0)$ 的估计值，可以通过合理猜测或基于历史数据的统计推断得到。由于 α 和 β 的值， $\hat{\mu}(0)$ 和 $\hat{T}(0)$ 初始值的影响将在一些实际观测值参与计算后“淡出”。

由于使用了带线性趋势的指数平滑，这个系统也可用于图示改进的过程。趋势 $\hat{T}(n)$ 是产能提升的良好指示器。若它为正值，平均产出在提升。在产多少就可以卖多少的环境中，较高的平均产能可以满足较高的产量定额，并因而带来较高的利润。

回忆起第十三章中计算经济生产定额时需要常规时间产量的均值 μ （和标准差 σ ）。我们也可以用指数平滑来追踪这个参数。方差比均值有更多的噪音，较难清晰地刻画趋势。处于这个原因，我们提倡使用无趋势的指数平滑。

令 Y_n 表示常规时间产量的第 n 个观测值， $\hat{\mu}(n)$ 表示常规时间产能的第 n 个平滑估计值， γ 为平滑常数。回忆起随机变量 X 方差的定义是

$$Var(X) = E[(X - E(X))^2]$$

第 n 个观测值之后，已经估计出常规时间产能的均值为 $\hat{\mu}(n)$ 。因此，我们可以估计第 n 个观测值之后的常规时间产能的方差为

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

这些估值中可能有噪音，所以使用先前的估值进行平滑，得常规时间产量方差的第 n 个估值

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

依照指数平滑的惯例，需要初值 $\hat{\sigma}^2(0)$ 来启动迭代。其后，每个新的常规时间产出观测值生成一个新的产量方差估计值。如在第十三章所见，较小的方差是我们能够较接近平均产能地设置定额，因而带来较大地利润。所以，趋势向下的 $\hat{\sigma}^2(n)$ 可作为逐步改善的生产系统

的指标。(479|480)

现在通过表 14.1 中的例子说明上述计算过程。常规时间段为周一到周五（每天两班），已按每周的产出收集了过去 20 周的数据。作为一个粗略的始点，我们乐观地估计产能为 2,000 件/周，故设定 $\hat{\mu}(0)=2,000$ 。不确定有趋势，故设定 $\hat{T}(0)=0$ 。猜测常规时间产量的标准差在 100 左右，故设定 $\hat{\sigma}^2(0)=10,000$ 。平滑常数选择为

$$\alpha = 0.5$$

$$\beta = 0.2$$

$$\gamma = 0.4$$

当然了，如第十章中的讨论，选择平滑常数在某种程度上也是一门艺术，需要在过去数据的基础上试错来得到合适的值。

表 14.1 产能参数的指数平滑

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	1400	1,700.0	-60.0	42,000.0	204.9
2	1302	1,471.0	-93.8	36,624.4	191.4
3	1600	1,488.6	-71.5	26,938.6	164.1
4	2100	1,758.5	-3.2	62,801.1	250.6
5	1800	1,777.7	1.2	37,880.4	194.6
6	2150	1,964.4	38.4	36,500.0	191.0
7	2450	2,226.4	93.1	41,898.8	204.7
8	2200	2,254.7	72.1	26,337.7	162.3
9	2600	2,463.4	99.4	23,263.2	152.5
10	2100	2,331.4	53.2	35,382.6	188.1
11	2200	2,292.3	34.7	24,636.7	157.0
12	2600	2,463.5	62.0	22,235.7	149.1
13	2800	2,662.7	89.4	20,877.2	144.5
14	2300	2,526.1	44.2	32,973.8	181.6
15	2900	2,735.2	77.2	30,653.1	175.1
16	2800	2,806.2	76.0	18,407.1	135.7
17	2650	2,766.1	52.7	16,433.0	128.2
18	3000	2,909.4	70.9	13,142.7	114.6
19	2750	2,865.1	47.8	13,188.1	114.8
20	3150	3,031.5	71.5	13,531.0	116.3

现在开始平滑的过程。第一期的常规时间产量是 1,400 件，所以使用式（14.3）计算出平均常规时间产能的平滑估计值（480|481）

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

类似地，使用式（14.4）计算平滑趋势

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

最后，使用式（14.5）计算常规时间产量方差的平滑估计值

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

所以，常规时间产量标准差的平滑估计值为 $\hat{\sigma}(1) = (42,000)^{1/2} = 204.9$ 。

继续这样的方式，产生表 14.1 的数据。将其绘制成图形，可以方便地进行检视。图 14.17 比较了常规时间产量的平滑估计值与实际值。注意到平滑估计值遵循了数据的上升趋势，但在各期之间有着较少的变动性（毕竟称为平滑）。还有，显然可以看出常规时间产能最初估计为 2,000 件/周有些高了。作为补偿，平滑趋势在最初几期朝下，直到实际的上升趋势最后迫使它抬高。

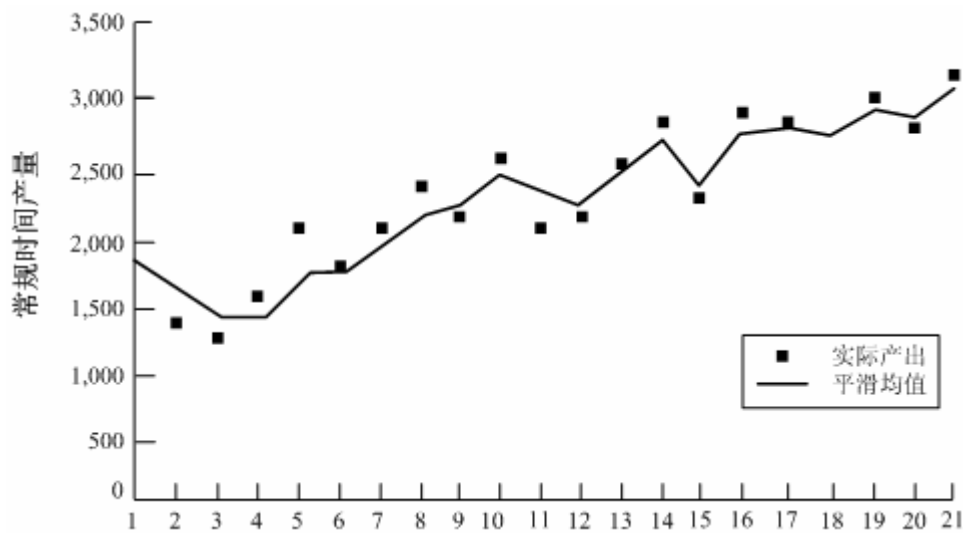


图 14.17 常规时间产能均值的指数平滑

这些趋势可以在绘出各期平滑趋势的图 14.18 中直接看到。由于高估了初值 $\hat{\mu}(0)$ ，趋势最初是负值。最后的正值趋势说明该厂的产能在提升，也是改善对运营显示效果的征兆。

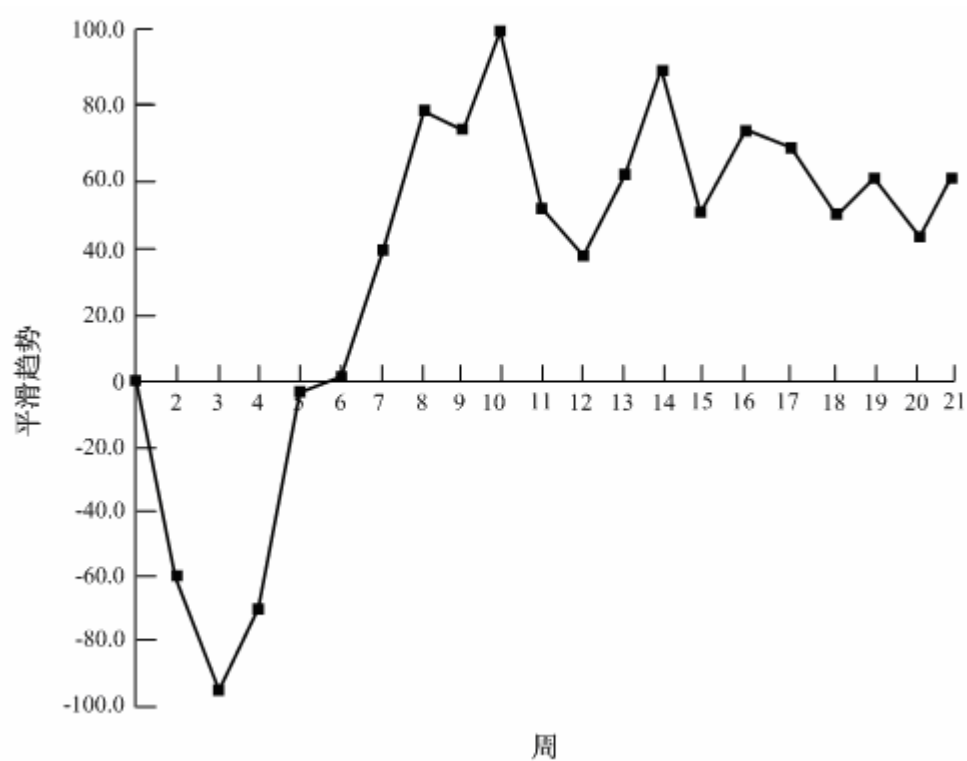


图 14.18 常规时间产能均值的指数平滑趋势

最后，图 14.19 绘出常规时间产量标准差的平滑估计值。这个估计值似乎为常数或略微下降。下降则说明该厂的改善正在削减产出变动性。标准差和平滑趋势都提供了持续改善的确信量度。(481|482)

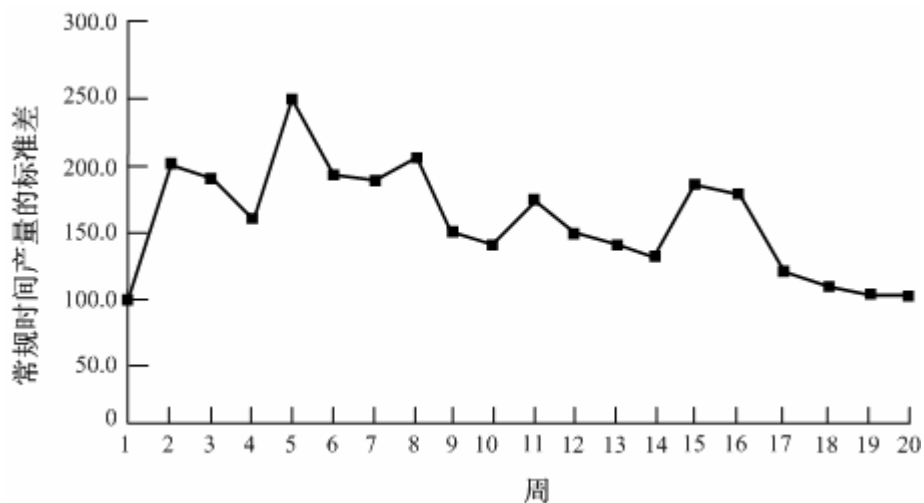


图 14.19 常规时间产能标准差的指数平滑

14.6 结论

本章中，我们花费了大量的时间来讨论生产计划与控制（PPC）体系中的车间作业控制

(SFC) 模块。我们强调, 良好的 SFC 模块的职能远多于简单地控制进出工厂的物料流动。作为连接制造流程的最低层级, SFC 在解决必须面对的管理问题时发挥了重要作用。设计良好的 SFC 模块将建立起可预测的、稳健的系统, 并以合乎要求的复杂度进行控制。

制造系统都是不同的, 所以适用于各种情形的通用 SFC 模块的想法不切实际。对非常广泛的情形足够通用的模块, 对简单系统显得赘余, 对特定的复杂系统显得不足。与 PPC 层级的其他模块不同, SFC 模块最要求定制化。(482|483) 可以使用商用条形码、光学扫描、局域网、统计过程控制以及其他技术, 作为 SFC 模块的组件。然而, 没有什么能够替代牢记系统的能力和需要并做出的慎重综合。我们希望, 读这本书的制造业专家以在这里或他处形成的基础知识、直觉以及综合技术能够实现这种综合。

我们不相信可以提供一步到位的方法 (cookbook scheme) 来设计合适的 SFC 模块, 所以我们的方法始于简单系统, 突出关键主题, 再扩展到各种较为复杂的主题。我们的基本想法始于一组简单的 CONWIP 产线, 并质问为何换模不可行。如果可行, 如我们相信它在相对简单的加工车间可行, 则这个就是最简单、最稳健的答案。如果不可行, 就需要从瓶颈处拉 (PFB) 等较复杂的设计。希望先前提出的 CONWIP 变种能激发读者创造性地为已讨论情形之外的具体问题追寻合理的解答。

最后要强调的主题是, 反馈是一个有效的生产计划与控制体系的本质特征。不幸的是, 许多 PPC 系统分布式地发展, 不同的群体对计划过程的不同方面负责。结果就是使用的数据不一致; 决策制定者之间的通讯中断; 面对问题时总是小团体主义和相互指责, 而不是合作与协调。还有, 若没有反馈机制, 过于乐观的数据 (如, 对产能不切实际的高估) 留存于计划系统, 其结果的最好情况是靠不住的, 最差则完全是滑稽可笑的。统计产出控制是一种推动所需的产能数据反馈的清晰机制。还可以设计类似方法来促进其他关键数据, 如制程产出、重工频率以及新产品的学习曲线等, 的反馈。关键是使管理人员对不一致的可能保持敏感, 并努力使 PPC 层级的反馈系统化。还有, 为了有效, 反馈机制的使用必须本着解决问题而非问责 (blame fixing) 的精神。

尽管 SFC 模块执行的是制造工厂中一些最为缓慢和平凡的任务, 它对于系统的整体有效性起着重要作用。设计良好的 SFC 模块建立了可预测的环境, 在其之上才能构筑其余的计划层级。合适的反馈机制可以为计划收集有用的数据, 并推动环境的持续改善。回忆起我们在本章开始时引用的格言

千里之行, 始于足下。
——老子

SFC 模块不仅是朝向有效的生产计划与控制体系的第一步, 事实上它也是非常重要的一步。

附录 14A 统计产出控制

解决几个短期生产追踪问题所需的基本数量是, 假设知道截止到现在已经生产的量, 在常规时间的末尾完成定额的可能性。(483|484) 由于各条产线的输出都被记录从而维持产线的恒定 WIP 水平, CONWIP 产线有所需的数据在手来做这些计算。

我们定义常规时间的长度为 R , 并假设这段时间的产量 N_R 服从正态分布, 均值 μ 标准差为 σ 。令 N_t 表示 $[0, t]$ 时间内的产量, 以标准单位计, $t \leq R$ 。模型中把 N_t 作为连续的、服

从正态分布的变量，均值为 $\mu t / R$ 方差为 $\sigma^2 t / R$ 。一般来说，产量服从正态分布的假设往往是适合的，除了 t 值较小的情形。这里对 N_t 均值和方差的假设等同于认为非重叠间隔内的产量相互独立。再一次地，这也很可能是个良好的假设，除了间隔较小的情形。

我们对 $N_t - S_t$ 最感兴趣，其中 S_t 是 $[0, t]$ 时间内产线的累积计划产量。如果使用了周期性的生产定额，则 $S_t = Qt / R$ 。 $N_t - S_t$ 表示 t 时刻过量生产，或超前于进度表的产量。它为正，我们超前；它为负，我们落后。在有着恒定生产速率的理想系统，这个值将总是零。在实际系统，它将有波动，变成正值和/或负值。

依据上述假设， $N_t - Qt / R$ 服从正态分布，均值为 $(\mu - Q)t / R$ 方差为 $\sigma^2 t / R$ 。类似地， N_{R-t} 也服从正态分布，均值为 $\mu(R-t) / R$ 方差为 $\sigma^2(R-t) / R$ 。故而，如果在 t 时刻， $N_t = n_t$ ，其中 $n_t - Qt / R = x$ （我们超前于进度 x 件），则仅当 $N_{R-t} < Q - n_t$ 时将在 R 时刻完不成定额。因此，给定当前的过量 x ， R 时刻完不成定额的概率为

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

其中 $\Phi(\cdot)$ 表示标准正态分布。

从实际执行的立场，预先计算引起完不成定额的概率为特定值 α 会更方便。表示如下：

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

得到

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

其中 z_α 对应于 $\Phi(z_\alpha) = \alpha$ 。这个 x 就是引起完不成定额的概率为 α 的 t 时刻过量值，也是 STC 图的基础——式 (14.2)。
