

文章编号:1006-5911(2002)11-0851-07

产品开发过程并行度优化

汪 峥,严洪森

(东南大学自动化研究所,江苏 南京 210096)

摘 要:并行工程环境中,上游产品设计活动和下游过程设计活动之间的重叠,可缩短产品开发过程的持续时间。但过多的重叠也会导致不必要的设计修改。为此,研究了产品与过程设计活动之间的并行度优化问题,并建立了数学模型,给出了求解该问题的算法,其目的是使因延迟产品开发完成时间和不必要的设计修改而带来的成本尽可能最低。

关键词:并行工程;设计活动;重叠;并行度优化

中图分类号:TP15

文献标识码:A

0 引言

在并行工程产品开发模式中,上游产品设计活动和下游过程设计活动之间存在进度上重叠。重叠可缩短产品开发过程的持续时间^[1],但过多的重叠,会增加设计修改工作量。因此,有必要对设计活动之间的并行度进行优化^[2,3]。为优化并行度,需定量描述与分析并行设计活动。文献[4]把上游设计活动看作一个进化过程,分析了下游设计对它的敏感性,指出若上游设计活动进化较快,下游设计活动可较早开始;反之,则晚些开始。文献[5,6]通过平均工程更改率进行不确定性建模,考虑了上下游活动之间的依赖关系,指出较快地消除不确定性,产品开发过程将从重叠中获利较多;反之,则获利较少。

当上游设计到一定阶段时,将已完成的设计预先发布,下游设计者即可对其进行评审,以便及时发现上游设计的错误并进行修改,该过程称为“设计迭代”。文献[7]研究了并行的产品与过程设计活动过程中的最优评审策略。文献[8]研究了通过选择最优评审频度,评审接受水平与设计工作强度来管理

产品开发过程中风险的方法。文献[4,5]研究了最优评审策略及设计迭代的特征。然而,他们仅强调上游设计修改对下游设计的影响,而下游设计活动发现上游设计错误的能力却未充分考虑。基于设计结构矩阵及其推广的“工作转移矩阵”,研究了多个设计活动之间的设计迭代特征^[9,10],讨论了设计活动完全串行^[9]或完全平行^[10]的情形,并根据工作转移矩阵的特征值和特征向量进行了设计模态分析^[10],但未考虑设计活动之间的重叠。

针对上述未研究的状况,笔者提出一种新的设计活动组模型来描述并行产品开发过程。设计活动组由一个上游产品设计活动和多个下游过程设计活动构成。该模型既强调上游设计活动对下游设计活动的影响,也注意描述下游设计活动发现上游设计活动的错误而导致的设计迭代,它描述的是上下游设计活动之间存在重叠的情形。本文给出了计算设计活动组持续时间和设计活动工作量的方法。在此基础上,提出设计活动并行度优化问题,其目标是通过优化下游过程设计活动的开始时间,来使设计活动组因延迟完成时间和不必要的设计修改而增加的成本尽可能最低,最后给出求解并行度优化问题的

收稿日期:2001-12-20;修订日期:2002-03-12。

基金项目:国家自然科学基金资助项目(69884001)。

作者简介:汪 峥(1973-),男,江苏南京人,现为上海交通大学计算机集成制造研究所博士后,主要从事并行工程、敏捷制造、生产系统设计与控制等研究。E-mail:zwang@cims.sjtu.edu.cn。

算法和实例。

1 设计活动组持续时间计算

在并行工程中,产品设计活动及其相关过程设计活动是并行的,设计评审与修改应当作为一个整体来考虑。由此给出如下定义:

定义1 产品设计活动及其相关过程设计活动通过设计迭代相互联系,构成设计活动组。

设计活动组由一个上游产品设计活动0和多个相关下游过程设计活动1, ..., m构成(如图1)。令 d_i ($i=0, 1, \dots, m$)表示所有设计活动的正常设计工作量;令 b_i 和 f_i 分别表示不考虑设计迭代时设计活动 i 的开始与结束时间。过程设计活动只有在产品设计活动获得结果后才进行。由于存在重叠,过程设计活动在产品设计活动全部完成之前获得的信息,只是部分完成的产品设计结果。因此,过程设计活动的进展受到产品设计活动进展的约束。

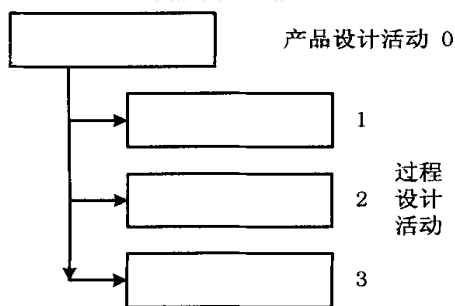


图1 设计活动组

假设1 产品设计活动和过程设计活动是连续匀速进行的,若产品设计活动已完成工作量占全部工作量的比例为 $([0, 1])$,则可完成的最大过程设计工作量占过程设计活动总工作量的比例也为

。

上述假设,使得本文的结果仅适用于技术成熟的产品开发过程或改型设计过程,而不适用于全新的产品开发过程。在产品创新设计中,不可预测的难点会导致设计过程的非线性;而在技术成熟的产品开发过程中,技术难点较少,因而可假设其设计过程是匀速的。实际产品开发中,大多数是利用成熟技术在现有产品的基础上进行改型设计。根据假设1,可给出产品设计活动对过程设计活动的制约函数的定义(如图2)。

定义2 令 x_0 和 x_i 分别表示产品设计活动0和过程设计活动 i 的已完成工作量,定义函数 f_{0i} : $x_0 \rightarrow x_i, x_0 \in [0, d_0], x_i \in [0, d_i]$,这里, $x_i = f_{0i}$

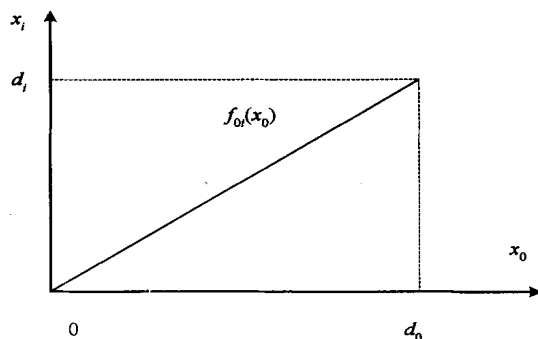


图2 制约函数

(x_0)意味着若产品设计活动已完成工作量为 x_0 ,则相应的过程设计活动能完成的工作量的上界为 x_i 。函数 f_{0i} 称作产品设计活动0对过程设计活动 i 的制约函数。

$$f_{0i}(x_0) = x_0 d_i / d_0 \quad (1)$$

令 $x_0(t)$ 和 $x_i(t)$ 分别表示产品与过程设计活动在时刻 t 已完成工作量,令 $b_0=0$,由假设1及定义2可知 $x_0(t)=t, x_i(t)=t-b_i$,且 $(t-b_i)/d_i \leq t/d_0$ 成立。若 $d_0 \leq d_i$,上式显然成立,并有 $f_i = b_i + d_i$;若 $d_0 > d_i$,则在时刻 $t^* = b_i d_0 / (d_0 - d_i)$ 时,上式中等号成立,此时产品与过程设计活动的进展相等。若 $b_i \geq d_0 - d_i$,则 $t^* \geq d_0$,此时产品设计活动已完成,过程设计活动能获得全部产品的设计结果,并可一直进行到它自身完成为止,即 $f_i = b_i + d_i$;若 $b_i < d_0 - d_i$,则 $t^* < d_0$,在时刻 t^* 时,过程设计活动必须停下来,等待进一步的产品设计结果,这就是产品设计活动对过程设计活动进度的制约;若 $b_i = d_0 - d_i$,则 $t^* = d_0$,此时产品设计活动和过程设计活动同时完成。于是过程设计活动 i 的最早开始时间为:

$$b_i = \begin{cases} b_0 = 0, & \text{若 } d_i \leq d_0 \\ d_0 - d_i, & \text{若 } d_i > d_0 \end{cases} \quad (2)$$

关于设计评审和修改,假设如下:

假设2 在产品设计活动进行过程中,在每两个时间区间 t 之间进行一次设计评审。设每次设计评审的时间相同,并用 Δt 表示。当评审进行时,产品和过程设计活动都应暂停,每次设计评审之后都可能触发设计修改。所有设计活动结束后,将进行最后一次设计评审,产品和过程设计活动也将进行最后一次修改。

令 p_i 表示第一次评审之前,产品设计结果不满足过程设计活动 i ($i=1, \dots, m$)要求的概率,于是第一次评审之前,产品设计结果不满足至少一个过

程设计活动要求的概率为 $p = 1 - \prod_{i=1}^m (1 - p_i)$ 。在每次评审和修改之后,这一概率将会降低,为此,给出如下定义:

定义 3 令 $g(u, v)$ 表示在产品设计进程的 u 点发生的错误在其进程的 v 点进行评审之后,仍然存在的概率与首次评审之前此错误发生的概率之比。对某一确定的 u , $g(u, v)$ 是 v 的非增函数,可称为“错误概率衰减函数”,它具有如下形式:

$$g(u, v) = a[(v - d_0)/(u - d_0)]^b + c \quad (3a)$$

或

$$g(u, v) = a\{ - [(v - u)/(d_0 - u)]^b + 1 \} + c \quad (3b)$$

式中: $v \in [u, d_0]$, $a > 0$, $b > 0$, $c > 0$ 及 $0 < a + c < 1$

式(3a)是凹的(图 3a),式(3b)是凸的(图 3b)。它们分别相应于文献[4]中快速和慢速进化的情形。于是,产品设计进程 u 点发生的错误在 v 点评审后仍存在的概率为 $pg(u, v)$ 。

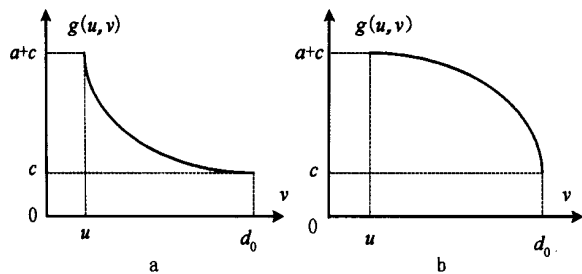


图3 错误概率衰减函数

由于产品设计进程中,每隔 t 时间有一次设计评审,因此,设计进程被划分为 $D = \text{ceil}(d_0/t)$ 个阶段,这里函数 $\text{ceil}(\cdot)$ 表示取不小于括弧内数的最小整数。假定每个过程设计活动都在某一阶段的起始点开始,则由式(2)可得过程设计活动 i 的最早开始阶段为:

$$X_i = \begin{cases} 2 & \text{若 } d_i \geq d_0 \\ \text{ceil}[(d_0 - d_i)/t] + 1 & \text{若 } d_i < d_0 \end{cases} \quad (4)$$

假定设计活动组仅由产品设计活动 0 和过程设计活动 i 构成。若过程设计活动 i 在阶段 X_i 开始,则活动组完成时间最早。若过程设计活动 i 在阶段 $x_i (> X_i)$ 开始,则活动组结束时间将会延迟 $(x_i - X_i)d$ 。因此过程设计活动 i 较早开始有利于设计活动组较早结束。但由于产品设计结果的不确定性,设计修改工作量将增加,而这种浪费应当最小化。

现在计算产品与过程设计活动修改工作量(如图 4)。假设过程设计活动 i 在阶段 x_i 的起始点开始,于是,在当前阶段 n 每个设计活动的已完成工

作量为:

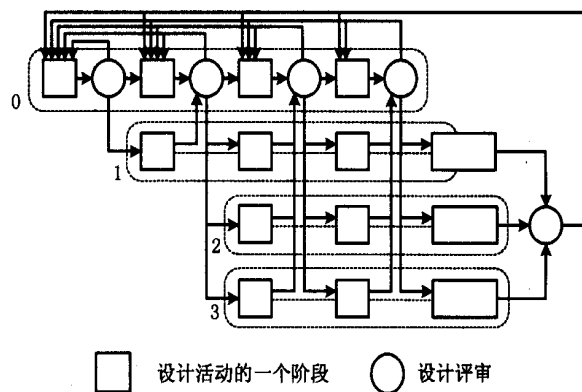


图4 设计活动组的运动过程

$$c_i(n) = \begin{cases} n \cdot t & \text{若 } i = 0 \\ 0 & \text{若 } i = 1, \dots, m \text{ 且 } n < x_i \\ (n - x_i + 1) \cdot t & \text{若 } i = 1, \dots, m \text{ 且 } n \geq x_i \end{cases} \quad (5)$$

令 p_{kn} 表示产品设计进程中第 k 阶段发生的错误在第 n 阶段被发现的概率,则有:

$$p_{kn} = \begin{cases} p & \text{若 } k = n \\ pg(k, t, (n - 1) \cdot t) & \text{若 } k < n \end{cases} \quad k = 1, \dots, n \quad (6)$$

令 $i \in [0, 1]$ 表示过程设计活动 i 发现产品设计的错误,而导致的设计迭代对所有设计活动修改工作量的影响程度,则总影响程度可定义为: $\rho = \max\{\rho_i, i = 1, \dots, m\}$ 。令 $w(n)$ 表示第 n 阶段的设计迭代持续时间均值, $w^{(0)}(n)$ 表示第 n 阶段产品设计活动 0 的设计修改工作量均值。第 n 阶段后的设计评审可能发现第 n 阶段及其以前各阶段的错误。于是:

$$w(n) = \sum_{k=1}^n t \cdot p_{kn} \quad (7)$$

产品设计活动的修改工作量为 $w^{(0)}(n) = w(n)$ 。若 $n \geq x_i$, 则设计活动 i 在阶段 n 的终点已完成工作量为 $c_i(n) = (n - x_i + 1) \cdot t$ 。根据制约函数, 决定了 $c_i(n)$ 的产品设计活动 0 的已完成工作量为 $c_0(n, i) = d_0 c_i(n) / d_i = d_0 (n - x_i + 1) \cdot t / d_i$, 其已完成的工作量覆盖了 $K(n, x_i) = \text{ceil}[c_0(n, i) / t] = \text{ceil}[d_0 (n - x_i + 1) / d_i]$ 个阶段, 其修改工作量为:

$$w^{(0)}(n, x_i) = \left\{ \sum_{k=1}^{K(n, x_i)-1} t \cdot p_{kn} + p_{K(n, x_i), n} [c_0(n, i) - (K(n, x_i) - 1) \cdot t] \right\} \quad (8)$$

相应地,过程设计活动 i 已完成部分的修改工作量为:

$$w^{(i)}(n, x_i) = d_i w^{(0)}(n, x_i) / d_0 \quad (9)$$

因此,在产品设计活动进程中,过程设计活动 i 的总修改工作量为:

$$w^{(i)}(x_i) = \sum_{n=x_i}^D w^{(i)}(n, x_i) = t \frac{d_i}{d_0} \sum_{n=x_i}^D \sum_{k=1}^{K(n, x_i)-1} p_{kn} + p_{K(n, x_i), n} [d_0(n - x_i + 1)/d_i - K(n, x_i) + 1] \quad (10)$$

在所有过程设计活动的设计工作量完成之后,要
对产品设计活动进行最后一次评审与修改。令 q_k
表示最后一次评审中产品设计活动第 k 阶段的错误被发
现的概率,则当 $k < D$ 时, $q_k = pg(k, t, d_0)$; 当 $k = D$
时, $q_k = pg(d_0, d_0) = p$ 。令 $[0, 1]$ 表示最后一次设
计迭代对产品和过程设计活动修改工作量的影响程
度,若产品设计活动 0 的阶段 k 有错误,则受影响范围
是从阶段 k 到阶段 D , 修改工作量为 $d_0(k) = [d_0 -$
 $(k-1)t]$ 。过程设计活动 i 受影响的已完成部分的
修改工作量为: $d_i(k) = d_i d_0(k)/d_0 = d_i [d_0 - (k-1)$
 $t]/d_0$ 。令 $y = \max\{x_i + \text{ceil}(d_i/t) - 1, i = 1, \dots,$
 $m\}$, 表示设计活动组设计工作的完成阶段, 则 $y - t$ 即
为最后一次设计迭代的开始时间。令 $t = t$ 为把
最后一次设计修改进程划分为若干阶段的最小时间单
位, 则产品设计活动最后一次设计修改开始阶段为 X_0
 $= 1$ 。类似于式(4), 过程设计活动 i 的最后一次修改
开始阶段为: 若 $d_i(k) \geq d_0(k)$, 则 $X_i(k) = 2$; 若 $d_i(k)$
 $< d_0(k)$, 则 $X_i(k) = \text{ceil}[(d_0(k) - d_i(k))/t] + 1$ 。
于是最后一次修改的持续时间为: $w(k) = \max\{(X_i$
 $(k) - 1)t + d_i(k), i = 0, 1, \dots, m\}$ 。在最后一次修改
中, 产品设计活动任何阶段的错误都有可能被发现, 于
是每个设计活动的修改工作量均值为: $d_i = \sum_{k=1}^D \sum_{l=1}^{k-1} q_l q_k d_i(k)$, 最后一次设计迭代持续时间均值为: w
 $= \sum_{k=1}^D (1 - q_l) q_k w(k)$ 。由此可知, 最后一次设计迭
代持续时间均值不依赖于 x_i 。

现在可计算整个设计活动组的持续时间, 以及各
设计活动的总工作量。根据式(7), 在所有设计活动的
设计工作完成之前, 产品设计活动 0 的修改工作量为:
 $w^{(0)} = \sum_{n=1}^D w^{(0)}(n) = \sum_{n=1}^D w(n)$ 。于是设计活动组的
持续时间均值为: $W = y - t + (D+1)t + w^{(0)} + w$ 。产
品设计活动 0 的总工作量均值为: $W^{(0)} = d_0 + (D+1)t$
 $+ w^{(0)} + d_0$ 。过程设计活动 i ($i = 1, \dots, m$) 的总工
作量均值为: $W^{(i)} = d_i + (D - x_i + 2)t + w^{(i)}(x_i) + d_i$ 。

2 并行度优化

产品与过程设计活动之间并行度优化问题, 首先

考虑的是仅由两个设计活动构成的设计活动组, 产品
设计活动 0 和过程设计活动 i 。并行度优化的目标是
通过优化过程设计活动 i 的开始阶段 x_i , 使设计活动
组因延迟完成时间和不必要的修改工作量而增加的成
本尽可能最小。

令 C_1 表示设计活动组完成时间延迟一个阶段
(t) 而增加的成本, C_2 表示每个时间单位的修改工
作量而增加的成本。根据式(7), $w^{(0)}(n)$ 不依赖于 x_i , 因
此, 为使不必要的修改工作量而增加的成本尽可能最
小时, 只需考虑过程设计活动 i 的修改工作量。于是并
行度优化问题的目标函数为:

$$\min C(x_i) = C_1(x_i - X_i) + C_2 w^{(i)}(x_i) \quad (11)$$

式中 $X_i \in [x_i, D]$, 首先给出下面的引理(证明略)和定
理。

引理 1 对于区间 $[X_i, D]$ 内的任意三个点 $x_i < x_j$
 $< x_k$,

$$C(x_j) < C(x_i) + (x_j - x_i) [C(x_k) - C(x_i)] / (x_k - x_i) \quad (12)$$

成立的充分必要条件是对任意三个相邻点 $x_i, x_i + 1$ 和
 $x_i + 2$, 下式成立

$$C(x_i + 1) < [C(x_i + 2) + C(x_i)] / 2 \quad (13)$$

定理 1 设计活动组的成本函数 $C(x_i)$ 是 x_i 的单
峰函数, 即区间 $[X_i, D]$ 内有唯一极值点。

证明: 显然, 仅需证明成本函数 $C(x_i)$ 是凹的, 即
对任意三个点 $x_i < x_j < x_k$, 必有 $C(x_j) < C(x_i) + (x_j$
 $- x_i) [C(x_k) - C(x_i)] / (x_k - x_i)$ 。由引理 1 可知, 仅
需证明任意三个相邻点 $x_i, x_i + 1$ 和 $x_i + 2$, $C(x_i + 1)$
 $< [C(x_i + 2) + C(x_i)] / 2$ 成立。

$$\begin{aligned} C(x_i + 2) + C(x_i) - 2C(x_i + 1) = \\ C_2 \{ w^{(i)}(x_i + 2) - w^{(i)}(x_i + 1) + w^{(i)}(x_i) - \\ w^{(i)}(x_i + 1) \} = C_2 t \frac{d_i}{d_0} \{ \sum_{n=x_i+2}^D \sum_{k=1}^{K(n, x_i+2)-1} p_{kn} + \\ \sum_{n=x_i}^D \sum_{k=1}^{K(n, x_i)-1} p_{kn} + p_{K(n, x_i), n} [d_0(n - x_i - 1)/d_i - K(n, x_i) + \\ 2] + 1 \} - \sum_{n=x_i+1}^D \sum_{k=1}^{K(n, x_i+1)-1} p_{kn} - \sum_{n=x_i+1}^D \sum_{k=1}^{K(n, x_i+1)-1} p_{K(n, x_i+1), n} \\ [d_0(n - x_i)/d_i - K(n, x_i + 1) + 1] + \\ \sum_{n=x_i}^D \sum_{k=1}^{K(n, x_i)-1} p_{kn} + \sum_{n=x_i}^D \sum_{k=1}^{K(n, x_i)-1} p_{K(n, x_i), n} [d_0(n - x_i + \\ 1)/d_i - K(n, x_i) + 1] - \sum_{n=x_i+1}^D \sum_{k=1}^{K(n, x_i+1)-1} p_{kn} - \end{aligned}$$

$$D$$

$$p_{kn}(n, x_i+1), n[d_0(n - x_i)/d_i - K(n, x_i+1) + 1] \quad (14)$$

若函数 p_{kn} 是凹的,即采用式(3a)的形式,由式(14)可得:

$$C(x_i+2) + C(x_i) - 2C(x_i+1) = C_2 \quad t \frac{d_i}{d_0}$$

$$D \quad K(n, x_i+1) - 1 \quad K(x_i+1, x_i+1) - 1$$

$$\{ - \quad p_{kn} - \quad p_{k, x_i+1} +$$

$$D \quad K(n, x_i) - 1 \quad K(x_i, x_i) - 1 \quad D$$

$$n = x_i+1 \quad k = K(n, x_i+1) \quad p_{kn} + \quad p_{k, x_i} + \quad p_{K(n, x_i+2), n}$$

$$[d_0(n - x_i - 1)/d_i - K(n, x_i+2) + 1] +$$

$$D$$

$$p_{K(n, x_i)}, n[d_0(n - x_i+1)/d_i - K(n, x_i) +$$

$$n = x_i \quad D$$

$$1] - 2 \quad p_{K(n, x_i+1)}, n[d_0(n - x_i)/d_i - K(n, x_i+1) + 1] \quad (15)$$

对上式中的第 1, 5, 7 项,分别令 $n = m+1, n = m+2$ 和 $n = m+1$,变换后用 n 代替 m 。由于 $K(n+a_1, x_i+a_2) = \text{ceil}[d_0(n+a_1-x_i-a_2+1)/d_i] = K(n, x_i+a_2-a_1)$,则上式可变换为:

$$C(x_i+2) + C(x_i) - 2C(x_i+1) = C_2 \quad t \frac{d_i}{d_0}$$

$$D-1 \quad K(n, x_i) - 1 \quad K(x_i, x_i) - 1$$

$$\{ (p_{kn} - p_{k, n+1}) + (p_{k, x_i} -$$

$$K(D, x_i) - 1 \quad D-1$$

$$p_{k, x_i+1}) + \quad p_{KD} + (p_{K(n, x_i), n+2} +$$

$$k = K(D, x_i+1) + 1 \quad n = x_i$$

$$p_{K(n, x_i), n} - 2p_{K(n, x_i), n+1} [d_0(n - x_i+1)/d_i -$$

$$K(n, x_i) + 1] + p_{K(D, x_i), D} [d_0(D - x_i+1)/d_i -$$

$$K(D, x_i) + 1] + p_{K(D, x_i+1), D} - p_{K(D, x_i+1), D+1}$$

$$[d_0(D - x_i)/d_i - K(D-1, x_i) + 1] \quad (16)$$

根据函数 p_{kn} 的定义, $p_{kn} - p_{k, n+1} > 0$ 和 $p_{k, x_i} - p_{k, x_i+1} > 0$ 成立。由于 $p_{K(n, x_i), n} > p_{K(n, x_i), n+1} > p_{K(n, x_i), n+2}$ 成立,及函数 p_{kn} 是凹的,则 $p_{K(n, x_i), n+2} + p_{K(n, x_i), n} - 2p_{K(n, x_i), n+1} > 0$ 成立。因为 $p_{K(D, x_i+1), D} > p_{K(D, x_i+1), D+1}$, 以及 $d_0(D - x_i)/d_i - K(D-1, x_i) + 1 = d_0(D - x_i)/d_i - \text{ceil}[d_0(D - x_i)/d_i] + 1 - 1$ 成立,则有 $p_{K(D, x_i+1), D} - p_{K(D, x_i+1), D+1} [d_0(D - x_i)/d_i - K(D-1, x_i) + 1] > 0$, 又因为 $d_0(n - x_i+1)/d_i - K(n, x_i) + 1 = d_0(n - x_i+1)/d_i - \text{ceil}[d_0(n - x_i+1)/d_i] + 1 > 0$, 所以 $d_0(D - x_i+1)/d_i - K(D, x_i) + 1 =$

$d_0(D - x_i+1)/d_i - \text{ceil}[d_0(D - x_i+1)/d_i] + 1 > 0$ 及 $p_{kn} > 0$ 成立,根据式(16)可得:

$$C(x_i+2) + C(x_i) - 2C(x_i+1) > 0 \quad (17)$$

对于函数 p_{kn} 为凸(即采用式(3b)的形式)的情形,可类似地证明式(17)成立,于是式(13)成立。因此,成本函数 $C(x_i)$ 是凹单峰函数,它在定义域 $[X_i, D]$ 内有且仅有一个极值点,证毕。

若定义域 $[x_i, D]$ 内整数点的数量 $D - X_i + 1$ 不多,则可用枚举法求得 $C(x_i)$ 的最小值。反之,由于 $C(x_i)$ 的极值点的唯一性(定理 1),可根据下述算法获得此极值点(用 x_i^* 表示之),其算法步骤如下:

步骤 1 初始化,若 $D - X_i + 1 = 3$,计算 $C(X_i)$, $C(X_i+1)$ 和 $C(D)$,选择 $x_i^* \in \{X_i, X_i+1, D\}$ 满足 $C(x_i^*) = \min\{C(X_i), C(X_i+1), C(D)\}$,则极值点为 x_i^* , $C(x_i^*)$ 为相应的成本函数值,转步骤 5;若 $D - X_i + 1 > 3$,则令 $x_i^{(0)} = X_i$ 及 $x_i^{(1)} = D$ 并转步骤 2。

步骤 2 令 $x_i^{(2)} = \text{ceil}[(x_i^{(0)} + x_i^{(1)})/2] - 1$ 及 $x_i^{(3)} = \text{ceil}[(x_i^{(0)} + x_i^{(1)})/2]$,计算 $C(x_i^{(0)})$, $C(x_i^{(1)})$, $C(x_i^{(2)})$ 和 $C(x_i^{(3)})$ 。

步骤 3 因为 $C(x_i)$ 有唯一极值点,则 $C(x_i^{(0)}) > C(x_i^{(2)}) > C(x_i^{(3)})$ 或 $C(x_i^{(2)}) < C(x_i^{(3)}) < C(x_i^{(1)})$ 必有一个成立;若 $C(x_i^{(0)}) > C(x_i^{(2)}) > C(x_i^{(3)})$ 成立,则 $x_i^{(2)} < x_i^* < x_i^{(1)}$ 一定成立,因为若 $x_i^* < x_i^{(2)}$,则根据 $C(x_i)$ 的单峰特性,必有 $C(x_i^{(2)}) < C(x_i^{(3)})$,于是产生矛盾;令 $x_i^{(0)} = x_i^{(2)}$,同理,若 $C(x_i^{(2)}) < C(x_i^{(3)}) < C(x_i^{(1)})$ 成立,则 $x_i^{(0)} < x_i^* < x_i^{(3)}$ 一定成立,令 $x_i^{(1)} = x_i^{(3)}$,若 $x_i^{(1)} - x_i^{(0)} + 1 = 3$,则转步骤 4;否则转步骤 2。

步骤 4 计算 $C(x_i^{(0)} + 1)$,并选择 $x_i^* \in \{x_i^{(0)}, x_i^{(0)} + 1, x_i^{(1)}\}$ 满足 $C(x_i^*) = \min\{C(x_i^{(0)}), C(x_i^{(0)} + 1), C(x_i^{(1)})\}$,于是 x_i^* 为极值点, $C(x_i^*)$ 为相应的成本函数值。

获得了设计活动组内过程设计活动的最优开始阶段后,即可计算其结束阶段 $y_i^* = x_i^* + \text{ceil}(d_i/t) - 1$ 。当设计活动组内包含多个过程设计活动时, $y^* = \max\{y_i^*, i = 1, \dots, m\}$ 即为设计活动组的最优结束阶段, y^* 确定了由设计活动组结束时间延迟而增加的成本。

定理 2 $w^{(i)}(x_i)$ 是 x_i 的单调下降函数。

证明:根据式(10),令 $n = m+1$,并在变换后令 n 代替 m ,则可得:

$$w^{(i)}(x_i+1) - w^{(i)}(x_i) = t \frac{d_i}{d_0} \{ \quad D-1 \quad K(n, x_i) - 1$$

$$n = x_i \quad k = 1$$

$$\begin{aligned}
 & (p_{k,n+1} - p_{kn}) + \sum_{n=x_i}^{D-1} (p_{K(n,x_i),n+1} - p_{K(n,x_i),n}) [d_0 \\
 & (n - x_i + 1)/d_i - K(n, x_i) + 1] - \sum_{k=1}^{K(D,x_i)-1} p_{KD} - \\
 & p_{K(D,x_i),D} [d_0 (D - x_i + 1)/d_i - K(D, x_i) + 1] \} \quad (18)
 \end{aligned}$$

因为 $p_{k,n+1} < p_{k,n}$, $p_{K(n,x_i),n+1} < p_{K(n,x_i),n}$ 及 $d_0 (D - x_i + 1)/d_i - K(D, x_i) + 1 > 0$, 所以 $w^{(i)}(x_i + 1) - w^{(i)}(x_i) < 0$ 成立, 即 $w^{(i)}(x_i)$ 是单调下降函数, 证毕。

根据定理 2, 在设计活动组的结束时间确定后, 所有过程设计活动开始时间都可推迟, 以降低设计修改成本。因此, 各过程设计活动的最优开始时间为:

$$x_i^* = y^* - \text{ceil}(d_i/t) + 1, i = 1, \dots, m \quad (19)$$

为使设计活动组的总成本最小化, 所有的过程设计活动都在同一阶段 y^* 结束。这里, x_i^* 实际上意味着产品设计活动和过程设计活动之间的最优并行度。Haberle 等也讨论了并行产品开发项目中设计活动并行度的测量方法, 但并未涉及到并行度优化问题^[11]; Nicoletti 和 Nicolo 通过管理产品开发项目中的信息流, 来使设计活动之间的并行时间最大化, 但未考虑设计迭代^[12]。

3 算例

设有一复杂机械零件的设计活动组, 它包括零件结构设计活动 0, 加工工艺设计活动 1 和工装夹具设计活动 2。零件结构设计活动 0 是产品设计活动, 而另两个活动是过程设计活动, 其正常设计时间分别为 $d_0 = 100$, $d_1 = 80$ 和 $d_2 = 120$ 。令 $t = 5$ 为时间单位, 于是设计活动 0 的正常设计进程可划分为 $D = d_0/t = 20$ 个阶段, 在每个阶段的终点将进行设计评审, 每次评审时间为 $\tau = 1$ 。令设计活动 1 和 2 发现设计活动 0 的错误概率为 $p_1 = p_2 = 0.5$, 则首次评审后, 零件结构设计错误被发现的概率为 $p = 1 - (1 - p_1)(1 - p_2) = 0.75$ 。令设计活动 0 的错误概率衰减函数为 $g(u, v) = 0.60 [(v - 100)/(u - 100)]^4 + 0.20$, 则设由设计活动 1 或 2 发现零件结构设计的错误而导致的设计迭代对所有设计活动修改工作量的影响程度分别为 $\alpha_1 = 0.5$ 和 $\alpha_2 = 0.5$, 于是总影响程度为 $\alpha = \max(\alpha_1, \alpha_2) = 0.5$ 。

根据式(4), 由于 $d_1 < d_0$, 则工艺设计活动 1 的最早开始阶段为 $X_1 = \text{ceil}[(d_0 - d_1)/t] + 1 = \text{ceil}[(100 - 80)/5] + 1 = 5$; 由于 $d_2 > d_0$, 则工装夹具设计活动 2 的最早开始阶段为 $X_2 = 2$ 。现在根据算法 1, 以设计活动 1 为例, 来解释它与设计活动 0 之间的并行度优化

方法。

步骤 1 由于 $D - X_i + 1 = 20 - 5 + 1 = 16 > 3$, 于是令 $x_1^{(0)} = 5$ 及 $x_1^{(1)} = 20$, 并转步骤 2。

步骤 2 令 $x_1^{(2)} = \text{ceil}[(5 + 20)/2] - 1 = 12$ 及 $x_1^{(3)} = \text{ceil}[(5 + 20)/2] = 13$ 。现在来计算 $C(x_1^{(0)})$, 若 $n = 1$, 则由式(5)可得 $c_0(1) = t = 5$ 及 $c_1(1) = 0$, 在此阶段的评审之后, 只有阶段 1 的错误能被发现, 由式(6)可得 $p_{11} = p = 0.75$ 。于是由式(7)可得 $w^{(0)}(1) = tp_{11} = 0.5 \times 5 \times 0.75 = 1.875$, 而此时 $w^{(1)}(1, 5) = 0$ 。类似地可得, 当 $n = 2, 3, 4$ 时, 各设计活动的修改工作量。若 $n = 5$, 则由式(5)可得 $c_0(5) = 5 - t = 25$ 及 $c_1(5) = (5 - 5 + 1) - t = 5$, 由式(6)可得 $p_{15} = pg(t, 4 - t) = 0.376$, $p_{25} = 0.431$, $p_{35} = 0.503$, $p_{45} = 0.600$ 和 $p_{55} = 0.750$, 由式(7)可得 $w^{(0)}(5) = t \sum_{k=1}^5 p_{k5} = 6.651$ 。当 $n = 5$ 时, 决定了 $c_1(5)$ 的设计活动 0 的部分已完成工作量为 $c_0(5, 1) = 100 \times 5/80 = 6.25$, 它覆盖了 $K(5, 5) = \text{ceil}[100 \times (5 - 5 + 1)/80] = 2$ 个阶段, 则由式(8)可得 $w^{(0)}(5, 5) = 0.5 \times 5 \times [p_{15} + p_{25} [100(5 - 5 + 1)/80 - 2 + 1]] = 1.209$, 由式(9)可得 $w^{(1)}(5, 5) = 80 \times 1.209/100 = 0.968$ 。类似地可得, 当 $n = 6, \dots, 20$ 时, $w^{(1)}(n, 5)$ 的值。根据式(10), 可计算出设计活动 1 的总修改工作量为 $w^{(1)}(5) = 79.969$ 。设成本系数为 $C_1 = 10.0$ 和 $C_2 = 1.0$, 则与设计活动 1 相关的总成本为 $C(x_1^{(0)}) = C_1(5 - 5) + C_2 w^{(1)}(5) = 79.969$ 。类似地可得 $C(x_1^{(2)}) = 87.294$, $C(x_1^{(3)}) = 93.708$ 和 $C(x_1^{(1)}) = 150.375$ 。

步骤 3 由于 $C(x_1^{(2)}) < C(x_1^{(3)}) < C(x_1^{(1)})$ 成立, 则令 $x_1^{(1)} = 13$; 又因为 $x_1^{(1)} - x_1^{(0)} + 1 = 13 - 5 + 1 = 9 > 3$, 因此转步骤 2。

继续进行下去直至 $x_1^{(0)} = 6$, $x_1^{(2)} = 7$, $x_1^{(3)} = 8$ 及 $x_1^{(1)} = 9$, 相应地 $C(x_1^{(0)}) = 70.078$, $C(x_1^{(2)}) = 67.296$, $C(x_1^{(3)}) = 68.665$ 及 $C(x_1^{(1)}) = 71.854$ 。根据步骤 3, 因为 $C(x_1^{(2)}) < C(x_1^{(3)}) < C(x_1^{(1)})$, 则令 $x_1^{(1)} = 8$; 又因为 $x_1^{(1)} - x_1^{(0)} + 1 = 8 - 6 + 1 = 3$, 于是转步骤 4。

步骤 4 因为 $C(7) = \min\{C(6), C(7), C(8)\}$, 所以可得 $x_1^* = 7$ 。

类似地可得 $x_2^* = 5$, 于是可得 $y_1^* = x_1^* + \text{ceil}(d_1/t) - 1 = 7 + \text{ceil}(80/5) - 1 = 22$ 及 $y_2^* = 28$, 于是 $y^* = \max\{y_1^*, y_2^*\} = 28$ 。根据式(19), 可得 $x^* = y^* - \text{ceil}(d_i/t) + 1 = 28 - \text{ceil}(80/5) + 1 = 13$ 及 $x_2^* = 5$, 它们分别为过程设计活动 1 和 2 的最优开始阶段。进一步

可求出设计活动组的持续时间为 $W = 370.105$, 设计活动 0, 1 和 2 的工作量分别为 $W^{(0)} = 318.496$, $W^{(1)} = 138.100$ 和 $W^{(2)} = 252.364$ 。

参考文献:

- [1] HAUPTMAN O, HIRJI K K. The influence of process concurrency on project outcomes in product development: an empirical study of cross-functional teams[J]. IEEE Transactions on Engineering Management, 1996, 43(2): 153 - 164.
- [2] AITSAHLIA F, JOHNSON E, WILL P. Is concurrent engineering always a sensible proposition? [J]. IEEE Transactions on Engineering Management, 1995, 42(2): 166 - 170.
- [3] YASSINE A A, CHELST K R, FALKENBURG D R. A decision analytic framework for evaluating concurrent engineering[J]. IEEE Transactions on Engineering Management, 1999, 46(2): 144 - 157.
- [4] KRISHNAN V, EPPINGER S D, WHITNEY D E. A model-based framework to overlap product development activities[J]. Management Science, 1997, 43: 437 - 451.
- [5] LOCH C H, et al. Communication and uncertainty in concurrent engineering[J]. Management Science, 1998, 44: 1032 - 1048.
- [6] TERWIESCH C, et al. Measuring the effectiveness of overlapping development activities[J]. Management Science, 1999, 45(4): 455 - 465.
- [7] HA A Y, et al. Optimal timing of reviews in concurrent design for manufacturability[J]. Management Science, 1995, 41: 1431 - 1447.
- [8] AHMADI R, WANG R H. Managing development risk in product development processes[J]. Operations Research, 1999, 47: 235 - 246.
- [9] SMITH R P, et al. Identifying controlling features of engineering design iteration[J]. Management Science, 1997, 43: 276 - 293.
- [10] SMITH R P, et al. A predictive model of sequential iteration in engineering design[J]. Management Science, 1997, 43: 1104 - 1120.
- [11] HABERLE K A, BURKE R J, GRAVES R. A note on measuring parallelism in concurrent engineering [J]. International Journal of Production Research, 2000, 38(8): 1947 - 1952.
- [12] NICOLETTI S, NICILO F. A concurrent engineering decision model: management of the project activities information flow [J]. International Journal of Production Economics, 1998, 54: 115 - 127.

Concurrency Optimization in Product Development Process

WANG Zheng, YAN Hong - sen

(Research Institute of Automation, Southeast Univ., Nanjing 210096, China)

Abstract: In concurrent engineering, the overlap between the upstream product design activity and the downstream process design activity can shorten the duration of the product development process. But too much overlap may cause some unnecessary design revision workloads. Therefore, the problem of concurrency optimization between the product and process design activities is studied, and the mathematical model is founded. The algorithm for solving this problem is proposed, whose aim is to minimize the total costs caused by the delay of the completion time of the product development and the unnecessary design revision workloads.

Key words: concurrent engineering; design activity; overlap; concurrency optimization

Received 20 Dec. 2001; Revised 12 Mar. 2002.

Foundation item: Project supported by the National Natural Science Foundation, China (Grant No. 69884001).

2003 年度期刊开始征订

《计算机集成制造系统—CIMS》期刊 2003 年度征订工作业已开始。明年本刊仍为月刊, 每月 30 日出版发行, 大 16 开, 每期 84 个印页, 订价 12 元, 全年 144 元, 邮发代号 82 - 289, 全国各地邮局均可订阅。也可直接与本刊编辑部联系订阅。

CIMS 编辑部