第 xx 卷第 x 期 控 制 理 论 与 应 用 Vol. xx No. x
xxxx 年 x 月 Control Theory & Applications Xxx. xxxx

文章编号: 1000−8152(2010)01−0000−00

最小化提前和延误惩罚的批处理问题

赵洪銮 韩国勇 张志军

(山东建筑大学计算机科学与技术学院, 山东 济南 250101)

摘要: 考虑机器容量有限的同时加工排序问题, 为享有公共交货期窗口 $[e, d]$ 的 n 个工件分批并排序以最小化总的赋权提前/延误惩罚. 本文把窗时排序与同时加工排序结合起来研究, 假设每个批的容量是 $b(< n$, 其中 n 为工件的个数), 而且最早交货期 e 和最晚交货期 d 已知. 但该问题是 NP- 完备的, 首先给出最优排序的几条性质, 进而解决了该问题的两类特殊情况.

关键词: 批; 交货期窗口; 提前; 延误

中图分类号: TP301, O157 **文献标识码:** A

Minimizing the earliness and tardiness penalty with batch processing

Zhao Hongluan, Han Guoyong, Zhang Zhijun

(School of Computer Science and Technology, Shandong Jianzhu University,
Shandong Jinan 250101, China)

Abstract: Common due window scheduling problem is concerned with batching to minimize the weighted earliness and tardiness penalties, where the jobs share a common due window [e,d]. Here, the due window situation combines with the conception of batch so that the jobs can be processed in batches. The bounded version of batch scheduling is considered such that at most b jobs can be processed in a batch with $b < n$, the number of jobs. Further, the location and size of due window are given parameters. The problem is NP-Complete. Based on optimal properties, two special cases are discussed.

Key words: batch; due window; earliness; tardiness

1 引言 (Introduction)

市场竞争越来越激烈, 为了避免隐藏的额外运转以及储存问题带来的高费用, 排序问题就不仅考虑延误带来的惩罚还必须顾及到提前完工付出的费用. 因此, 许多企业采用 “准时” 的观念以对生产做本质改进, 使得提前和延误的工件都要付出惩罚费用, 此类排序被称为*准时排序*. 假设工件的交货期是一个时间点. 如果工件在交货期之前完工, 则要将其储存在仓库, 从而就要付出与保管相关的费用; 若在其后完工, 则会产生延误赔偿甚至会失去合作机会. 而准时排序就是要最小化所有这些费用的和. 现实生产中的大部分交货期需要一定的公差容许量,“*交货期窗口*” 的概念则显得尤为重要, 它是一个被*最早交货期* e 和*最晚交货期* d 定义的时间区间, 其中*窗口大小*为 $K = d - e$. 显然, 这个模型更具有一般性和广泛的现实意义, 相应的排序称为*窗时排序*, 它的特殊形式是窗口大小为零的准时排序.

在过去的近二十年里, 关于提前和延误惩罚已经有很多结果, 参见综述 [1], 但集中在准时排序的讨论中; 关于窗时排序的文章不多而且大部分是关于公共交货期窗口. 真正引入 “交货期窗口” 概念的是 [2], 考虑提前和延误时间的惩罚. 当交货期窗口的位置待定但无费用的时候, 他给出了多项式时间算法; 而交货期窗口给定且最晚交货期小于总加工时间时, 问题是 NP- 完备的并给出拟多项式时间算法. [3, 4] 也讨论了同样的问题, 只是针对交货期窗口的位置或大小是已知还是待定而论.Liman 和 Ramswamy [5] 研究最小化赋权提前时间惩罚和延误工件个数的排序问题, 证明了限制型和非限制型都是 NP- 完备的并给出拟多项式时间的动态规划算法. 文献 [6] 研究最小化最大费用的窗时排序问题, 其中交货期窗口待定, 针对单机及多台机器展开讨论. 谭芳 [7] 探讨宽惩罚系数依赖于工件的情况, 但问题是 NP- 完备的, 并给出伪多项式算法. [8] 不仅考虑提前/延误惩罚, 还考虑附加惩罚, 在确定了最优公共交货期的基础上给出了相应的最优排序.

收稿日期: xxxx−xx−xx; 收修改稿日期: xxxx−xx−xx.

基金项目: 国家自然科学基金 (60972052), 山东省科技发展计划 (2010GGX10129).

另一方面, 批处理调度也成为热点问题, 分为两类. 其一是工件在机器上加工之前需要安装任务, 把工件分成多个类似的工件组, 当不同组的工件接连加工时, 则需要一定的安装任务; 把此类排序称为成组分批排序. 另一类也像经典排序一样不考虑安装时间和安装费用, 但是可以在一台机器上同时加工多个工件, 即所谓的同时加工排序; 这两类排序问题都比较难于研究, 涉及的大部分问题是 NP- 困难的. 本文研究的是后者, 并与窗时排序问题结合起来考虑. 同时加工排序问题主要在于提高生产效率, 同时加工排序最早出现在半导体生产过程中, 后来在其他生产领域, 如冶金、电镀等得到应用, 因而具有广泛的应用价值和现实意义. 比较典型的是发生在大规模集成电路的生产中. 为了保证成品合格, 在电路的最后测验阶段, 往往把集成电路放在烤箱中试验它们能否在一定的温度下经受一定时间的烘烤.

一个*批*是指同时被加工且同时完工的工件集合, 当这个批中的所有工件完成时才称此批完工. 一旦一个批开始加工, 既没有其他工件加入其中, 批中任一个工件也不能被取走. 因此, 批的加工时间等于这个批中最长的工件加工时间. 所以, 同时加工排序问题首先把工件分成多个批然后再排列这些批的次序, 使得某个目标函数最大或者最小.

同时加工排序问题的研究结果参见综述 [9]. 当所有工件有共同的就绪时间及 m 个不同的加工时间时, 对 m 固定的情况,Hochbaum 和 Landy [10] 提出了运行时间为 $O(m^2 3^m)$ 的最优算法以最小化总完工时间; 而当 m 不固定时, 他们给出了 2- 近似算法.Brucker 等人 [11] 得到了 $O(n^{b(b-1)})$ 时间的动态规划算法; 当有 m 个不同加工时间时, 它的动态算法所需时间为 $O(b^2m^2 2^m)$. 进而, 证明了几个准时排序问题是 NP- 困难的; 并说明当批的个数固定时, 任意正则函数都有多项式时间算法. 后来, [12] 对有相同或任意就绪时间的总完工时间问题给出多项式时间近似方案 (PTAS). 马建辉等 [13] 研究提前/拖期惩罚的批调度问题, 不仅考虑了提前/拖期惩罚, 还考虑了机器的加工费用. 在给出批调度优化的 4 个性质的基础上提出了两个启发式算法. 本文作者在文献 [14] 中首次把窗时排序与同时加工排序结合起来讨论, 但其中批的容量是无界的. 而本文研究的是有界同时加工排序问题.

2 问题描述及最优性质 (Presentation and properties)

要在一台批处理机上加工 n 个相互独立且不可中断的工件 $\{J_1, J_2, \cdots, J_n\}$, 所有工件同时就绪. 机器每次至多加工 b 个工件且 $b < n$. 符号 p_i, s_i, c_i 分别记工件 J_i 的加工时间, 开工时间和完工时间 $(i = 1, 2, \cdots, n)$, 类似地, 对批 B_j, 其加工时间 $P_j = \max\{p_i | \ J_i \in B_j\}$, 开工时间 $S_j = s_i$, 完工时间 $C_j = c_i$. 假设工件按 SPT 序标记下标使得 $p_1 \le p_2 \le \cdots \le p_n$.

所有工件享有公共交货期窗口 $[e, d]$, 其中 $d \ge e$, 窗口大小为 $K = d - e$. 本文假设 K 和 e 均给定. 不失一般性, 假设 $K \ge p_n$ 以及 $K < p_n + p_{n-b} + p_{n-2b} + \cdots + p_{n-\lfloor \frac{n}{b} \rfloor b}$, 否则, 所有工件都能在区间 $[e, d]$ 内完工. 如果工件 J_i 在交货期窗口之前完成, 会遭受提前惩罚, 提前时间定义为 $E_i = \max\{0, e - c_i\}$; 若在其后完工, 也要付出延误代价, 延误时间为 $T_i = \max\{0, c_i - d\}$. 进而, 定义排序 σ 的费用函数

$$Z(\sigma) = \sum_{i=1}^{n} (\alpha E_i + \beta T_i),$$

其中 α 和 β 分别是提前和延误惩罚系数. 定义提前集合、准时集合和延误集合分别为

$$E(\sigma) = \{B_j | \ C_j < e\}, \quad W(\sigma) = \{B_j | \ e \le C_j \le d\}$$

和 $$T(\sigma) = \{B_j | \ C_j > d\}.$$

在不引起混淆的情况下, 将它们分别记为 E, W 和 T. 既然一个批中的工件有相同的完工时间, 也定义这些工件属于它的批所在的集合.

文献 [2] 中证明了当 $b = 1$ 时, 此问题是 NP- 完备的. 推广可知当批容量 b 有限时, 最小化 $\sum\limits_{i=1}^{n} (\alpha E_i + \beta T_i)$ 的窗时排序问题也是 NP- 完备的.

下面提出最优排序的几条性质, 有助于探索问题的最优解. 显然, 在任何一个最优排序中, 从第一个批被加工至最后一个批的完工之间没有空闲时间.

性质1. *在最优排序中, 存在某些批, 使其在 e 或者 d 完工, 除非第一个批的开工时间为零.*

证明. 利用反证法, 假设存在一个最优排序 σ, 它的第一个批既不在零时刻开工, 排序中也没有在 e 或者 d 完工的批. 也就是, 存在批 B_i 和 B_j 使得 $S_i < e$ 且 $C_i > e, S_j < d$ 且 $C_j > d$.

情况 1. 把排序中的批序列向右移动 $\delta_1 = \min\{e - S_i, d - S_j\}$, 得到新排序 σ'. 如果 $\delta_1 = e - S_i$, 则 B_i 之前的批恰好在 e 完成; 若 $\delta_1 = d - S_j$, 则 B_j 之前的批在 d 完成. 于是总费用的变化为

$$\triangle Z = Z(\sigma') - Z(\sigma) = -\alpha |E(\sigma)| \delta_1 + \beta |T(\sigma)| \delta_1,$$

其中 $|E(\sigma)|$ 和 $|T(\sigma)|$ 分别记 $E(\sigma)$ 及 $T(\sigma)$ 中的工件个数.

情况 2. 令 $\delta_2 = \min\{C_i - e, C_j - d, S_1\}$, 其中 S_1 表示第一个批的开工时间. 将整个排序向左移动 δ_2,

并记新的排序为 σ''. 讨论同上, 针对 δ_2 的三个不同取值, 批 B_i 在 e 完工或者 B_j 在 d 完工, 或者第一个批的开工时间为零. 从而费用变化为

$$\triangle Z = Z(\sigma'') - Z(\sigma) = \alpha|E(\sigma)|\delta_2 - \beta|T(\sigma)|\delta_2.$$

所以, 如果 $\alpha|E(\sigma)| > \beta|T(\sigma)|$, 根据情况 1, 右移排序会减少惩罚, 并且得到 $S_i = e$ 或者 $S_j = d$. 否则, 把整个排序向左移动 δ_2 使得 $C_i = e$ 或者 $C_j = d$, 除非第一个批的开工时间为零; 这些均与 σ 的最优性矛盾, 即性质中的结论成立. □

从这个性质可以看出, 在一个排序中可能存在这样的批: 开工时间小于 e(或 d) 而完工时间大于 e(或 d), 相应地称它们为*跨越 e(或 d) 的批*.

性质2. *在最优排序中, 准时集合包含加工时间最小的那些工件.*

证明. 假设在一个最优排序 σ 中, 准时批 B_1 中的工件 J_k 及非准时批 B_2 中的工件 J_j 满足 $p_k > p_j$, 其中 $1 \le k, j \le n$ 且 $k \ne j$.

首先假设 $J_j \in E(\sigma)$. 若 $|B_1| < b$(其中 $|B_1|$ 表示批 B_1 中的工件个数), 把 J_j 放入批 B_1 后使得总费用减少. 否则, 交换 J_k 和 J_j, 把新的排序记为 σ'.

情况 1: B_2 中某个工件的加工时间不小于 p_k. 于是 B_2 的加工时间不变, 而 B_1 的加工时间可能变小. 如果 P_1 变小, 向左移动 B_1 后面的批使整个排序中没有空闲时间, 有 $\Delta Z = Z(\sigma') - Z(\sigma) \le 0$.

情况 2: B_2 中的每个工件均比 J_k 小. 不失一般性, 假设 J_j 是 B_2 中最大者, 而 J_k 是 B_1 中最大的. 类似地, 把 B_1 与 B_2 之间的批右移得到 $\Delta Z \le 0$. 检验两个批中的其它工件后, 要么结论成立, 要么出现情况 1.

反复进行以上操作, 直到最小的工件都移入交货期窗口中, 得到的排序费用减少, 与 σ 的最优性矛盾. 所以对任意工件 $J_k \in W(\sigma)$, $J_j \in E(\sigma)$, 必然有 $p_k \le p_j$. 如果 $J_j \in T(\sigma)$, 交换 J_k 和 J_j 的位置亦会改善排序性能. 因此,$W(\sigma)$ 包含加工时间最小的那些工件. □

如果一个批恰好包含 b 个工件, 称其为*满的*; 否则为*不满的*. 在文献 [14] 中批容量无限的情况, 最优排序是 SPT- 批序的. 那么对本文批容量有限的情形, 此结论是否成立呢?

性质3. *最优排序 $\sigma = \{B_1, B_2, \cdots, B_m\}$ 是 SPT- 批序的, 其中 $B_l = \{J_i, \cdots, J_j\}$, $1 \le l \le m$, $1 \le i \le j \le n$. 进而, 提前批按加工时间非增的顺序排列, 而延误批则是非降的顺序.*

证明. 根据性质 2, 最小的那些工件在交货期窗口中加工, 为了使占用的总时间最少, 它们一定满足 SPT- 批序而且加工时间较大的批都是满的, 只可能最小的不满. 因此, 我们只需考虑提前集合和延误集合中的工件. 用反证法验证它们满足 SPT- 批序.

情况 1. 假设提前批不满足 SPT- 批序, 即存在 $E(\sigma)$ 中的工件 $J_s, J_{s+2} \in B_k$ 及 $J_{s+1} \in B_t$, 其中 $1 \le k,t \le m$. 如果 $t < k$ 且 $|B_k| < b$, 把 J_{s+1} 放入 B_k 将减少总惩罚. 如果 $|B_k| = b$, 交换 J_{s+1} 与 J_{s+2}. 类似于性质 2 中的讨论, 新排序的惩罚函数值变小. 若 $k < t$, 交换 J_s 与 J_{s+1} 后也会出现同样的结果. 因此, 对任一个提前批, $l(B_k) \ge L(B_{k+1}), k = 1, 2, \cdots, m-1$, 其中 $L(B_k)$ 和 $l(B_k)$ 分别是 B_k 中工件的最大和最小加工时间.

对延误集合的工件, 同理可知, 前一个批中任一工件的加工时间不大于后一个批的加工时间.

情况 2. 设 J_s 和 J_{s+2} 是提前批 B_k 中的两个工件,J_{s+1} 是延误批 B_t 的成员. 如果 $P_t > p_{s+1}$, 交换 J_{s+1} 和 J_{s+2}. 则 P_k 不会增大而 P_t 不变. 如果 P_k 变小, 把批 B 前面的工件右移将使总惩罚变小. 如果 $P_t \le p_{s+1}$, 交换 J_{s+1} 和 J_s, 由于 P_t 减小, 通过类似讨论, 同样出现矛盾.

总之, 每个批均是包含下标相邻的工件, 而且提前批按加工时间非增的顺序排列, 而延误批则是非降的顺序. □

根据以上性质, 整个排序是 SPT- 批序的, 最优排序就可以被每个批中的最小工件确定, 于是, 把注意力集中在满足 SPT- 批序的排序上. 而对延误集合中的工件而言, 是要最小化它们的总完工时间, 结合 [15] 中对总完工时间的讨论给出以下性质.

性质4. [15] *延误批序列 $B_1, B_2, \cdots, B_r$ 是最优序列当且仅当*

$$\frac{P_1}{|B_1|} \le \frac{P_2}{|B_2|} \le \cdots \le \frac{P_r}{|B_r|},$$

性质5. *提前集合中批序列 $(B_1', B_2', \cdots, B_r')$ 是最优的, 当且仅当*

$$\frac{P_1'}{|B_1'|} \ge \frac{P_2'}{|B_2'|} \ge \cdots \ge \frac{P_r'}{|B_r'|}.$$

如果某些工件的加工时间相等, 按其不同值分类, 设有 h 类, 使得 $p_1 < p_2 < \cdots < p_h$; 类 $t \in \{1, 2, \cdots, h\}$ 中有 n_t 个工件, 则 $n = \sum\limits_{t=1}^{h} n_t$. 如果一批中的所有工件来自于同一类, 称为*同类的*; 否则为*非同类的*. 若 $p_t = P_i$, 称 t *支配*批 B_i. 根据 [15] 中的结果,

性质6. *关于类 t, $t \in \{1,2,\cdots,h\}$, 最优排序中有 $\lfloor \frac{n_t}{b} \rfloor$ 个同类的满批; 这些批在提前集合、准时集合及延误集合中是相邻的.*

于是, 我们可以首先把类中的这些工件分配到同类的满批中, 剩下 $n_t - b\lfloor \frac{n_t}{b} \rfloor$ 个 t 类工件,$t \in \{1,2,\cdots,h\}$.

性质7. [15] *在任一个最优排序中, t 类至多支配一个非同类的批.*

3 特殊情况的求解 (Special cases)

3.1 如果 $E = \emptyset$ (If $E = \emptyset$)

在很多现实情况下, e 的取值比较小或者 $\alpha \gg \beta$, 则在最优排序中 $E = \emptyset$, 因此有必要讨论这种情况的最优解; 再者, 可以借鉴一些比较成熟的研究结果. 根据性质 1, 在一个最优排序中, 第一个批的开工时间为零, 或者某个批恰好在 e 或 d 完工. 当 $E = \emptyset$ 时, 此结论仍然成立. 另一方面, 只要在第一个批的完成时间不小于 e 的条件下, 应该尽力向左移动批序列以减少延误工件的惩罚. 根据性质 2, 准时集合包含加工时间最小的那些工件, 比较容易确定. 然后再对 $e = 0$ 和 $e > 0$ 两种情况, 最小化延误工件的总完工时间. 然而, 当 b 不是固定常数时, 最小化总完工时间的问题复杂性还没有解决, [12] 给出了多项式时间近似方案 (PTAS).

尽管 W 包含最小的一些工件, 但并不说明它包含最多工件就最优; 也就是说, 只要使 W 最小的批多包含一个工件 (不满时), 且所有准时批仍是 SPT-批序的,T 中的最小工件可能能被 W 容下; 但这会增加延误费用, 从而需要综合考虑. 因此, 准时工件的个数由最小批的工件个数及 K 的值所确定. 用 A 分别表示集合 E,W,T, 则 A 中的批序列被标记为 $B_1^A, B_2^A, \cdots$. 于是对 $E = \emptyset$ 的情况, 得到以下的 PTAS.

算法 3-1

步骤 3-1.1 按加工时间的 SPT 顺序标记工件使得 $p_1 \le p_2 \le \cdots \le p_n$. 令 $E = \emptyset, k = 1$.

步骤 3-1.2 记 $r = \min\{j | \sum_{f=0}^{j} p_{k+fb} \ge K\}$. 把 J 中最小的 k 个工件放入 B_1^W. 如果 $e + K \le \sum_{f=0}^{r} p_{k+fb}$, 置 $W = \{B_{r-1}^W, \cdots, B_1^W\}$ 使其满足 SPT 批序且较大批是满的, 令 $s = k + (r-1)b$ 及 $S(B_{r-1}^W) = 0$. 否则, 令 $W = \{B_r^W, \cdots, B_1^W\}$ 及 $s = k + rb$, 其中 B_r^W 为跨越批. 在保证 $C_r^W \ge e$ 的前提下最大限度地向左移动这些批.

步骤 3-1.3 对其他剩余工件, 按文献 [12] 中的方式放入延误集合 $T = J - W$ 中, 得到最小化它们的总完工时间 $C_{k,sum}$ 的 PTAS.

步骤 3-1.4 若 $k < b$, 令 $k := k + 1$ 并转至步骤 3-1.2. 否则, 转至步骤 3-1.5.

步骤 3-1.5 对 $k = 1,2,\cdots,b$, 根据步骤 3-1.2 中的两种情况计算 $Z = \beta(C_{k,sum} - (n-s)d)$, 选取惩罚值最小的作为最好的近似方案. □

定理1. *当 $E = \emptyset$ 时, 算法 3-1 给出了多项式时间近似方案 (PTAS).*

3.2 所有加工时间相等 (With equal processing time)

在本节, 假设 $p_1 = p_2 = \cdots = p_n = p$, 性质 1∼7 仍然成立.

性质8. *在一个最优排序中, 除了第一个批或最后一个批外, 其它都是满的.*

基于以上最优排序的性质, 根据性质 1 的结论分三种情况讨论最优算法.

程序 I: $\exists B$ 使得 $C(B) = e$

步骤 I.1 每 b 个工件作为一批装入 $B_1^W, \cdots, B_k^W$ 中, 其中 $k = \lfloor K/p \rfloor$ 且 $S_1^W = e$.

步骤 I.2 如果 $e \ge p$, 将足够多个工件放入 B 使得 $C(B) = e$ 且 $|B| \le b$, 转至步骤 I.3. 如果 $e < p$, 停止, 因为没有完工时间为 e 的批存在.

步骤 I.3 如果提前集合中的剩余时间小于 p, 其它工件将会放入延误集合使得前面批是满的, 转至步骤 I.6. 否则, 执行步骤 I.4.

步骤 I.4 用 $E_{B'}$ 和 $T_{B'}$ 分别表示把 B' 放在现有批序列之前和之后的提前时间和延误时间. 如果提前集合中的剩余时间足够大且 $\alpha E_{B'} < \beta T_{B'}$, 把足够多个工件放入批 B' 作为提前工件使得 $|B'| \le b$; 否则, 均作为延误工件.

步骤 I.5 对于其它工件, 执行步骤 I.3∼I.4, 即使剩余工件的个数小于 b.

步骤 I.6 调整排序以满足文中提到的性质, 计算总惩罚 Z_1.

程序 II: $\exists B$ 使得 $C(B) = d$

步骤 II.1 每 b 个工件作为一批放入 $B_1^W, \cdots, B_k^W$, 其中 $k = \lceil K/p \rceil$, $C_k^W = d$.

步骤 II.2 类似于步骤 I.3∼I.4 排列其它工件. 再计算总惩罚 Z_2.

程序 III: 第一个批的开工时间为零

步骤 III.1 如果 $e \leq p$, 转至步骤 III.3. 否则, 令 $l = 1$, $B_1^E = \{p_1, \cdots, p_l\}$ 使得 $S_1^E = 0$. 再把其它工件放入满批中, 最后一个批可能不满; 排在 B_1^E 后面.

步骤 III.2 若 $l < b$, 令 $l := l + 1$, 使得 $B_1^E = \{p_1, \cdots, p_l\}$, 转至步骤 III.1. 否则, 计算 $l = 1, 2, \cdots, b$ 时的总惩罚, 并把最小者记作 Z_3.

步骤 III.3 每 b 个作为一批排下去, 使整个排序无空闲时间且第一个批的开工时间为零. 计算总惩罚 Z_3'.

算法 3-2

执行程序 I-III 后, 得到并比较 Z_1, Z_2, Z_3 或 Z_3'. 选择有最小惩罚的排序作为最优排序. □

从算法 3-3 的进程来看, 它显然是最优的. 通过简单的时间计算, 程序 I 和程序 II 分别用时 $O(n)$, 而程序 III 所用时间为 $O(\max(bn, b\log b)) = O(bn)$. 因此

定理2. *当所有工件的加工时间相等时, 算法 3-2 在 $O(bn)$ 时间内得到最优排序.*

4 结语 (Conclusion)

研究有公共交货期窗口的同时加工排序问题, 而且批容量是有限的, 每个批至多同时加工 b 个工件, 目的是最小化总的赋权提前/延误惩罚. 通过分析, 得到最优排序的几条性质; 但是此问题是 NP- 完备的. 于是本文解决了两类特殊情况.

参考文献(References):

[1] Gordon V., Proth J.M., Chu C.. A survey of the state-of-the-art of common due date assignment and scheduling research[J]. *European Journal of Operational Research*, 2002, 139(1), 1-25.

[2] Kramer F. J., Lee C. Y.. Common due window scheduling[J]. *Production and Operations Management*, 1993, 2(2), 262-275.

[3] Liman S. D., Panwalkar S. S., Thong S.. Determination of common due window location in a single machine scheduling problem[J]. *European Journal of Operational Research*, 1996, 93(1), 68-74.

[4] Liman S. D., Panwalkar S. S., Thong S.. Common due window size and location determination in a single machine scheduling[J]. *Journal of the Operational Research Society*, 1998, 49(9), 1007-1010.

[5] Liman S. D., Ramswamy S.. Earliness-tardiness scheduling problems with a common delivery window[J]. *Operations Research Letters*, 1994, 15(4), 195-203.

[6] Mosheiov G., Sarig A.. Minmax scheduling problems with a common due-window[J]. *Computers and Operations Research*, 2009, 36(6), 1886-1892.

[7] 谭芳, 孙世杰. 宽容交货加权超前延误单机排序问题 [J]. *上海大学学报自然科学版*,2005, 11(2), 149-154.
(TAN Fang, SUN Shi-jie. Scheduling Jobs for a Common Due Window to Minimize Weighted Sum of Earliness and Tardiness Penalties[J]. JOURNAL OF SHANGHAI UNIVERSITY(NATURAL SCIENCE EDITION), 2005, 11(2), 149-154.)

[8] 吴悦, 汪定伟. 交货期窗口下带有附加惩罚的单机提前/拖期调度问题 [J]. *控制理论与应用*, 2000, 17(1), 9-13,18.
(WU Yue, WANG Dingwei. Single Machine Earliness/Tardiness Scheduling Problem with Additional Penalties about Due Window[J]. Control Theory & Applications, 2000, 17(1), 9-13,18.)

[9] Potts C. N., Kovalyov M. Y.. Scheduling with batching :A review[J]. *European Journal Of Operational Research*, 2000, 120(2), 228-249.

[10] Hochbaum D.S., Landy D.. Scheduling semiconductor burn-in problem operations to minimize total flowtime[J]. *Operations Research*, 1997, 45, 874-885.

[11] Brucker P., Gladky A., Hoogeveen H., et al. Scheduling a batching machine[J]. *Journal of Scheduling*, 1998, 1(1), 31-54.

[12] Deng X., Li G., Feng H., Shi B.. A PTAS for semiconductor burn-in scheduling[J]. *Journal of Combinatorial Optimization*, 2005, 9(1), 5-17.

[13] 马建辉, 牛海军. 提前/拖期惩罚的单机批调度优化问题研究 [J]. *制造业自动化*, 2002, 24(7), 65-67.
(MA Jianhui, NIU Haijun. The optimal batch scheduling with early and tardy penalty on single machine[J]. Manufacturing Automation, 2002, 24(7), 65-67.)

[14] Zhao H., Li G.. Unbounded batch scheduling with a common due window on a single machine[J]. Journal of Systems Science and Complexity. 2008, 21(2), 296-303.

[15] Chandru V., Lee C.Y., Uzsoy R.. Minimizing total completion time on a batch processing machine with job families[J]. *Operarional Research Letters*, 1993, 13(2), 61-65.

作者简介:

赵洪銮 (1979—), 女, 博士, 讲师, 主要研究方向为算法分析与设计, E-mail: hongluanzhao@163.com;

韩国勇 (1978—), 男, 硕士, 实验师, 主要研究方向为软件工程;

张志军 (1973—), 男, 硕士, 副教授, 主要研究方向为网络工程.