

考虑学习效应的最小化延误总时间的单机批次排序问题

胡金昌, 吴耀华[†], 吴颖颖, 杨 栋

(山东大学 控制科学与工程学院, 济南 250061)

摘要: 一些生产场景中, 工件以批次作业的形式被安排生产, 工件批量大、加工工序基本相同, 所以标准工时相同, 而且实际加工时间会受到学习效应的影响. 为此, 讨论学习效应的最小化延误总时间的单机批次排序问题, 对该问题建立数学模型. 该问题属于 NP-hard 问题, 采用动态规划算法 (DP) 和模拟退火算法 (SA) 求解该问题, 通过实验分析不同规模时 DP 的执行时间与 SA 的执行时间和求解误差的变化趋势, 比较 SA 与其他实践中常用的经典规则的求解效果. 最后得出 DP 适合批次数小于 13 的小规模问题, 可以得到精确解; 与经典规则相比, SA 至少可以使目标函数降低 20%, 表明 SA 算法具有有效性. SA 解决大规模问题时效果较优, 并得出 SA 的执行时间和误差随着控制参数改变的变化趋势.

关键词: 批次排序问题; 单机; 学习效应; 动态规划; 模拟退火算法; 最小化延误总时间

中图分类号: O223

文献标志码: A

Batch scheduling with learning effect on single-machine to minimize the total tardiness

HU Jin-chang, WU Yao-hua[†], WU Ying-ying, YANG Dong

(College of Control Science and Engineering, Shandong University, Jinan 250061, China)

Abstract: In some factory, job has been processed in batches which consist of the processing procedures basic identical jobs, so all jobs have the same normal processing time. The actual processing time is affected by the “learning effect”. The batch scheduling problem with learning effect on a single-machine to minimize the total tardiness is discussed, which is a NP-hard problem. For this problem, we model and propose a dynamic programming (DP) algorithm and a simulated annealing (SA) algorithm. The solution time variation trend of the both algorithms and the errors variation trend of SA with different problem sizes are analyzed by experimental tests. We have compared the performance of the SA to other classical rules which are usually applied in real-life. The results show that the DP can get exact solution and more suit the small size problems whose batch number is less than 13. Comparing to other classical rules, the SA is effective because of 20% decline in object value by using the SA. The large size problems can be solved better by using the SA, and the variation trend of the solution time and the errors with controlling parameters changed is analyzed.

Keywords: batch scheduling; single-machine; learning effect; dynamic programming; simulated annealing algorithm; minimizing the total tardiness

0 引言

目前, 国家大力推进传统产业优化升级, 生产企业在引进大批先进制造设备的同时, 也不断学习先进的管理理念和管理科学, 生产调度便是该范畴的内容. 通过优化调度, 可在不增加硬件成本的前提下, 通过优化生产排序提高企业生产效率, 节省成本, 一直以来得到了企业界和学术界的重视.

学习效应是实际生产排序问题中需要考虑的重要因素. 随着工人操作时间的增长, 工人不断重复同

样的工作, 对所从事的生产操作越来越熟悉, 工件加工或者作业的时间也会随时间变短, 这样的现象被称为学习效应^[1]. Biskup^[2] 和 Cheng 等^[3] 首先将学习效应应用在排序问题中. Biskup^[4] 总结了学习效应两种主要的数学模型: 依赖工件位置的学习效应模型和依赖完成时间的学习效应模型. 依赖工件位置的学习效应由 Biskup^[2] 提出, 是学习效应的经典模型, 得到了广泛应用. 依赖完成时间的学习效应由 Yang 等^[5] 首次提出.

收稿日期: 2018-05-05; 修回日期: 2018-08-12.

基金项目: 国家自然科学基金项目 (61703241).

责任编辑: 刘士新.

[†]通讯作者. E-mail: mike.wu@263.net.

批次生产存在两种场景:一种是平行批次生产,另一种是串行批次生产.目前考虑学习效应的批次生产研究多为平行批次生产^[6-8],本文所考虑的串行批次生产近些年才得到广泛研究. Pei等^[6]提出了考虑准备时间、学习效应、恶化效应的批次排序模型. Yusriski等^[9]提出了考虑学习效应、恶化效应的批次排序模型,其目标函数为总流水加工时间,并用启发式算法求解. Li等^[10]建立考虑释放时间和学习效应的批次排序问题,并用分支定界算法和遗传算法解决了最小化总完成时间问题. 这些考虑学习效应的串行批次排序的研究都是针对工件先分批后排序,而既定批次考虑学习效应的最小化总延误时间的批次排序问题尚未得到研究.

然而,在某些生产中,工厂以批次作业的形式加工工件,工件工序基本相同,所以标准工时相同,实际加工时间受到学习效应的影响. 例如制衣行业的缝衣车间,所有短袖T恤都是由左袖、右袖、前幅、后幅4片布料缝制而成,标准工时相同,衣服以订单的形式完成加工,一个订单中包含多件衣服,可视为一批. 既定批次的考虑学习效应的单机批次排序问题在实际生产中普遍存在,能够快速准确解决该问题,大幅提高生产效率,节约成本.

本文建立了考虑学习效应的单机批次排序问题最小化延误总时间的数学模型. 提出了动态规划算法(DP)和模拟退火算法(SA)求解该问题,设计不同规模实验分析了DP的执行时间及SA执行时间与误差的变化趋势,并证明了SA在求解大规模问题时的有效性,最后总结了两个算法的适用条件.

1 数学建模

1.1 问题描述及符号含义

一共有 N 批工件,用集合表示为 $A = \{B_1, B_2, \dots, B_N\}$,不同批次的工件数量、工期和权重不同,分别用集合表示为 $\{N_1, N_2, \dots, N_N\}$ 、 $\{D_1, D_2, \dots, D_N\}$ 和 $\{w_1, w_2, \dots, w_N\}$,所有工件的标准工时相同为 P ,都是0时刻到达工厂. 只有一个加工机器,并且同一时间只能完成一个工件,工件加工不能中断,工件之间不能存在时间间隔,所有批次依次完成,不能交叉. 工件的实际加工时间受依赖工件加工位置的学习效应的影响,目标是得到最小的批次总延误时间及批次加工顺序.

其他符号定义如下: $B_{[r]}$ 为第 r 批加工的批次; $N_{[r]}$ 、 $D_{[r]}$ 、 $w_{[r]}$ 、 $p_{[r]}$ 、 $C_{[r]}$ 、 $T_{[r]}$ 分别为 $B_{[r]}$ 的工件数量、工期、权重、实际加工时间、完成时间、延误时间; $P_j^{[r]}$ 为 $B_{[r]}$ 的第 j 个工件的实际加工时间; $S_{[r]}$

为第1批到第 r 批加工的工件总数; Z_b^r 为决策变量,第 r 批加工的批次为 b 时该变量取值为1,否则取值为0, $r = 1, 2, \dots, N, b = 1, 2, \dots, N$.

1.2 构建数学模型

用决策变量表示 $B_{[r]}$ 的工件数量、工期、权重,即

$$N_{[r]} = \sum_{b=1}^N Z_b^r N_b, \quad r = 1, 2, \dots, N; \quad (1)$$

$$D_{[r]} = \sum_{b=1}^N Z_b^r D_b, \quad r = 1, 2, \dots, N; \quad (2)$$

$$w_{[r]} = \sum_{b=1}^N Z_b^r w_b, \quad r = 1, 2, \dots, N. \quad (3)$$

根据依赖工件位置的学习效应模型^[2],工件 J_j 实际的工作时间 $p_{r,j} = t_j r^a, t_j$ 表示工件 j 的标准工时, $a \leq 0$ 代表学习因子, r 表示 J_j 被安排加工的位置. 所以 $B_{[r]}$ 的第 j 个工件的实际加工时间

$$P_j^{[r]} = P(S_{[r-1]} + j)^a, \quad r = 1, 2, \dots, N, j = 1, 2, \dots, N_{[r]}. \quad (4)$$

其中

$$S_{[r]} = \begin{cases} \sum_{i=1}^r N_{[i]}, & r = 1, 2, \dots, N; \\ 0, & r = 0. \end{cases} \quad (5)$$

用 $S_{[r]}$ 表示 $p_{[r]}$ 、 $C_{[r]}$ 、 $T_{[r]}$ ($r = 1, 2, \dots, N$) 为

$$p_{[r]} = \sum_{j=1}^{N_{[r]}} P_j^{[r]} = \sum_{j=1}^{N_{[r]}} P(S_{[r-1]} + j)^a, \quad (6)$$

$$C_{[r]} = \sum_{i=1}^r p_{[i]} = \sum_{i=1}^r \sum_{j=1}^{N_{[i]}} P_j^{[i]} = \sum_{i=1}^r \sum_{j=1}^{N_{[i]}} P(S_{[r-1]} + j)^a, \quad (7)$$

$$T_{[r]} = \max \{C_{[r]} - D_{[r]}, 0\}. \quad (8)$$

该问题模型为

$$\min \sum_{r=1}^N w_{[r]} T_{[r]}. \quad (9)$$

$$\text{s.t.} \sum_{b=1}^N Z_b^r, \quad r = 1, 2, \dots, N; \quad (10)$$

$$\sum_{r=1}^N Z_b^r, \quad b = 1, 2, \dots, N. \quad (11)$$

目标函数和约束条件可通过式(1)~(8)求解. 条件(10)限制了一次只能加工一个批次;约束条件(11)限制了一个批次只能被一次加工完成.

1.3 模型性质

定理1 任意批次完成时间 $C_{[r]}$ 与前 $r - 1$ 个批次的工件数量有关,与前 $r - 1$ 个批次加工顺序无关,

为一定值.

证明 将式(7)展开,有

$$C_{[r]} = P \sum_{j=1}^{N_{[1]}+N_{[2]}+\dots+N_{[r]}} j^a = P \sum_{j=1}^{S_{[r]}} j^a. \quad (12)$$

因为 $S_{[r]}$ 与批次加工顺序无关,所以 $C_{[r]}$ 值与批次加工顺序无关,等于 $P \sum_{j=1}^{S_{[r]}} j^a$. □

推论1 任意批次延误时间 $T_{[r]}$ 与第 $r+1$ 到第 N 批次的工件数量有关($r = N - 1, \dots, 1$),由第 $r + 1$ 到第 N 批所安排的批次集合决定,而与其加工顺序无关,为一定值.

证明 令 S 表示第 $r + 1$ 到第 N 批次的工件数量,所以 $S = N_{[r+1]} + N_{[r+2]} + \dots + N_{[N]}$.

$$T_{[r]} = \max \left\{ P \sum_{j=1}^{S_{[r]}} j^a - D_{[r]}, 0 \right\}, \quad (13)$$

其中 $S_{[r]} = N_{[1]} + N_{[2]} + \dots + N_{[r]} = S_{[N]} - S$.

式(13)中 P 、 $D_{[r]}$ 、 $S_{[N]}$ 为定值, S 与第 $r + 1$ 到第 N 批次的加工顺序无关,所以推理2得证. □

单机工件排序最小化总延误时间问题已经被证明是NP-hard问题^[11],所以本文问题也属于NP-hard问题.

2 动态规划算法

问题总共有 N 个批次需要完成,所以分为 N 个阶段,用 k 表示阶段, k 取值由 N 到1. 设 δ_k 为状态变量,代表第 k 阶段时,第 k 批到第 N 批加工的批次编号的集合,其中 $k = N, N - 1, \dots, 1, \delta_{N+1} = \phi$. 每阶段状态变量 δ_k 可能存在多种情况. 设 μ_k 为决策变量,代表第 k 阶段时第 k 批加工的批次编号.

$$\delta_k = \delta_{k+1} \cup \{\mu_k\}, \mu_k \notin \delta_{k+1}, \mu_k \in A. \quad (14)$$

设 $f_k(x)$ 代表第 k 阶段的决策函数, x 表示第 k 到第 N 共 $N - k + 1$ 个批次的批次编号集合. 状态转移方程为

$$\begin{cases} f_k(\mu_k) = W_k(\delta_{k+1}, \mu_k), k = N; \\ f_k(\delta_k) = \\ \min_{\mu_k \notin \delta_{k+1}, \mu_k \in A} \{f_k(\delta_{k+1}) + W_k(\delta_{k+1}, \mu_k)\}, k < N. \end{cases} \quad (15)$$

式(15)需满足式(14)中变量的关系. 由式(5)和(13)可知

$$W_k(\delta_{k+1}, \mu_k) = w_{\mu_k} T_{\mu_k} = w_{\mu_k} \max \left\{ P \sum_{j=1}^{S_{[N]}-S} j^a - D_{\mu_k}, 0 \right\},$$

其中 $S = \sum_{i \in \delta_{k+1}} N_i$.

设 $F_k(\delta_k)$ 代表第 k 阶段的决策,是使 $f_k(\delta_k)$ 取得最小值的 δ_k 集合中元素的顺序. $F_1(\delta_1)$ 为该问题的解, $f_1(\delta_1)$ 为目标函数值.

显然,式(14)表示第 k 阶段状态由第 $k + 1$ 阶段状态和第 k 阶段决策变量共同决定,同一状态 δ_k 可以是不同的 δ_{k+1} 和相应 μ_k 的组合. 决策函数 $f_k(x)$ 的作用是在不同 δ_{k+1} 和相应 μ_k 的组合中找到最优组合,并用 $F_k(\delta_k)$ 代表取得最优值时 δ_k 集合中元素的顺序.

例如3个批次的排序,当 $k = 2$ 时, $\delta_2 = \delta_3 \cup \{\mu_2\}$,此时 δ_2 分别为 $\{1, 2\}, \{1, 3\}, \{2, 3\}$. 当 $\delta_2 = \{1, 2\}$ 时存在两种组合方式, $\delta_3 = \{1\}, \mu_2 = \{2\}$ 和 $\delta_3 = \{2\}, \mu_2 = \{1\}$,根据式(15), $f_2(\{1, 2\}) = \min \left\{ \begin{matrix} f_3(\{1\}) + W_2(\{1\}, \{2\}) \\ f_3(\{2\}) + W_2(\{2\}, \{1\}) \end{matrix} \right\}$, 求出 $f_2(\{1, 2\})$,并用 $F_2(\{1, 2\})$ 记录 $f_2(\{1, 2\})$ 取得最大值的集合 $\{1, 2\}$ 的顺序,然后依次求出 δ_2 取 $\{1, 3\}$ 和 $\{2, 3\}$ 时的 f_2 和 F_2 ,第2阶段计算结束.

3 模拟退火算法

模拟退火算法是由初始解 X_0 和温度 $T = T_0$ 开始,对当前解重复“产生新解→计算目标函数值→接受或舍弃”的迭代,逐步减小 T 值,算法终止时的当前解即所得近似最优解,这是基于Monte Carlo迭代求解法的一种启发式随机搜索过程. 退火过程由冷却进度表控制,控制参数包括初始温度 T_0 、衰减因子 α 、每个 T 值时的迭代次数 L 和停止条件等.

定义该问题的解为批次的加工序列. 评价函数为

$$E(X) = \sum_{r=1}^N w_{[r]} T_{[r]}. \quad (16)$$

初始解:批次随机排序.

终止条件:一是温度衰减到终止温度 T_{end} 以下,即 $T < T_{\text{end}}$;二是加入阈值 θ_{end} ,同一目标函数值持续的次数 θ 超过 θ_{end} 时迭代终止,即 $\theta > \theta_{\text{end}}$.

产生新解的算法:随机生成两个序号,将两序号对应的两批次交换顺序.

模拟退火算法流程如下:

- 1) 初始化:设置初始解 X_0 、 T_0 、 L 、 T_{end} 、 θ_{end} 、 α ,令迭代次数 $i = 0, \theta = 0, X = X_0$.
- 2) 若温度 $T < T_{\text{end}}$,终止算法, X 为最终解;否则执行步骤3).
- 3) 判断 i ,若 $i \leq L$,执行步骤4);否则令 $i = 0$,

$T = \alpha T$,返回步骤2).

4)产生新解 X_{new} ;计算增量 $\Delta E = E(X_{new}) - E(X)$;若 $\Delta E < 0$,则接受 X 作为当前新的解,否则以概率 $\exp(-\Delta E/T)$ 接受 X 作为当前新的解.如果最后接受当前新的解,则令 $X = X_{new}, \theta = 0$;否则 $\theta = \theta + 1$.若 $\theta > \theta_{end}$,则终止算法, X 为最终解;否则令 $i = i + 1$,返回步骤3).

4 测试实验

为分析DP和SA的性能,本文设计了以下两组实验:1)小规模问题实验,目的:①衡量DP的执行时间和求解规模上限;②比较两种算法,得出DP在解决小规模问题的优势.2)大规模问题实验,目的:①与实际生产中常用的经典规则对比,验证SA的有效性;②衡量SA的求满意解的执行时间和误差,讨论控制参数对SA的影响.所有实验在Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz双核处理器、4.00GB的计算机上利用软件Matlab 2014 a完成.

4.1 小规模问题的实验

小规模问题是指批次数量 N 较小的问题.本文参考文献[12]设计了小规模测试实验.各批次的工件数量 N_r 满足 $[1, 100]$ 均匀分布,各批次工期 D_r 满足 $[0, C_{[N]}]$ 均匀分布,各批次权重 w_r 满足 $[0.5, 1]$ 均匀分布,工件标准工时 $P = 1$,学习因子 $a = -0.01$,批次数量 N 取 $5 \sim 18$ 的整数,对于不同的 N 进行10次实验取算法执行时间的平均值.

不同规模时该算法执行时间如图1所示,DP的执行时间会随批次数量 N 的增加急剧增长,当 $N = 17$ 时执行时间已接近3h,当 $N > 17$ 时,预计该算法的执行时间将超过一天,实际中是不能被接受的.

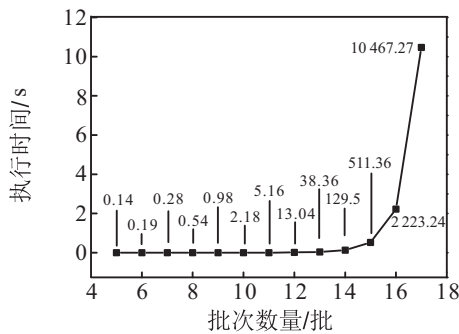


图1 不同规模时动态规划算法的执行时间

对于以上小规模实验,再用SA求解,用下式:

$$\text{误差} = (\text{SA解} - \text{DP解}) / \text{DP解} \quad (17)$$

计算SA的误差,误差可衡量SA求解准确性.

各个控制参数都会对算法的执行时间和误差产生影响,本文讨论了不同迭代次数、终止温度的

影响.算法的各个参数取值参考文献[13],即 $\theta_{end} = 500, \alpha = 0.995, T_0 = 500$. (L, T_{end}) 取组合值 $(1, 1), (1, 0.02), (50, 1), (50, 0.02)$.

由图2可得,相同参数时,SA求解的误差随着问题规模增大而变大,并可通过增加迭代次数或降低终止温度提高算法准确度.SA执行时间受参数影响较大,增加迭代次数或降低终止温度会增加执行时间,随着问题规模增加执行时间略有增加.

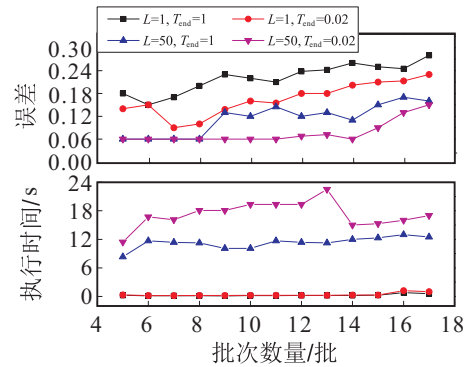


图2 小规模问题SA的误差和执行时间

对比图1和图2,当 $N < 13$ 时,若保证SA的准确度,需要令 $L = 50$,此时执行时间均大于DP;而DP可以得到精确解,所以DP较优.当 $N \geq 13$ 时,DP执行时间超过35s,SA都小于22s,并可通过调节参数,将执行时间和误差控制在可接受范围内,所以SA较好.

4.2 大规模问题的实验

大规模问题是指批次数量 N 较大的问题,利用DP求解该问题执行时间较长,适合用启发式算法得到近似最优解.本文测试的大规模实验是指批次数量 N 取 $100, 200, \dots, 1000$ 的整数的实验.

首先,本文设计了大规模问题情况下,SA与排序问题经典规则的比较,这些规则频繁应用于实际生产过程中:最短加工时间优先规则(SPT);加权最短加工时间优先规则(WSPT);最早工期优先规则(EDD);加权最早工期优先规则(WEDD).其中批次加工时间需忽略批次位置的影响,每批工件都从0时刻开始加工.图3展示了SA求解的目标函数值与应用经典规则结果的比较,可见应用SA在不同规模时至少可以使目标函数降低20%,具有应用的有效性.

然后,通过实验衡量了不同参数情况时SA求满意解的执行时间和误差,参数 $\theta_{end}, \alpha, T_0, L$ 和 T_{end} 组合与小规模实验参数取值相同.大规模实验不能得到问题的最优解,所以定义同一批次数量 N 的所有解中最优解为满意解,用式(18)计算其他参数所得解的误差.

$$\text{误差} = (\text{其他参数解} - \text{满意解}) / \text{满意解} \quad (18)$$

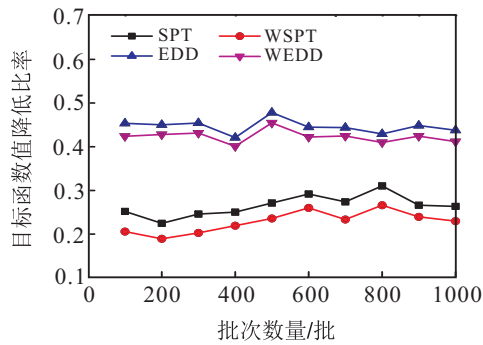


图3 SA相对其他方法求解目标函数值降低比率

图4为大规模问题SA的误差和执行时间。由图4可见:随着问题规模的增大,SA的执行时间也增长, L 值越大,增长越快; T_{end} 越小,增长越快,但是增长速度远低于DP,呈线性增长形式。 L 值越大或 T_{end} 越小,所得解越接近最优解,所以满意解几乎都是当 $L = 50, T_{\text{end}} = 0.02$ 时取得。参数确定的情况下,随着问题规模的增大,SA离最优解的偏离程度增大。通过观察两图中相同参数时的实验结果可以看出, L 对执行时间和误差的影响大于 T_{end} ,所以 L 适合对算法的执行和结果作大幅调节,而 T_{end} 适合细微调节。

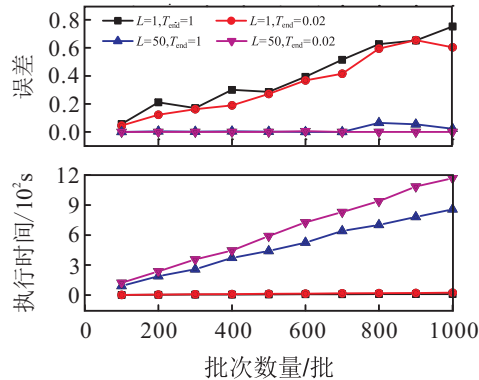


图4 大规模问题SA的误差和执行时间

5 结论

本文结合实际问题,提出了考虑学习效应的最小化延误总时间的单机批次排序问题,通过建模分析,提出了DP和SA两种方法解决该问题,并将问题划分为大规模和小规模问题分别作了实验分析。实验结果表明:在批次数量 $N < 13$ 时,选择DP较优;当批次数量 $N \geq 13$ 时,通过与经典规则比较,表明了SA的有效性;SA的执行时间可以接受,并且随问题规模增大的增长速度大幅减缓,选择SA更为合适;SA的求解精确度和执行时间可以通过改变参数调节,具有较强的灵活性。

参考文献(References)

[1] Badiru A B. Computational survey of univariate and multivariate learning curve models[J]. IEEE Trans on Engineering Management, 1992, 39(2): 176-188.

- [2] Biskup D. Single-machine scheduling with learning considerations[J]. European J of Operational Research, 1999, 115(1): 173-178.
- [3] Cheng T C E, Wang G. Single machine scheduling with learning effect considerations[J]. Annals of Operations Research, 2000, 98(1/2/3/4): 273-290.
- [4] Biskup D. A state-of-the-art review on scheduling with learning effects[J]. European J of Operational Research, 2008, 188(2): 315-329.
- [5] Yang D L, Kuo W H. Single-machine scheduling with an actual time-dependent learning effect[J]. J of the Operational Research Society, 2007, 58(10): 1348-1353.
- [6] Pei J, Liu X B, Panos M, et al. Serial-batching scheduling with time-dependent setup time and effects of deterioration and learning on a single-machine[J]. J of Global Optimization, 2017, 67(1/2): 251-262.
- [7] Qi X L, Zhou S G, Yuan J J. Single machine parallel-batch scheduling with deteriorating jobs[J]. Theoretical Computer Science, 2009, 410(8/9/10): 830-836.
- [8] Yang D L, Kuo W H. A single-machine scheduling problem with learning effects in intermittent batch production[J]. Computers & Industrial Engineering, 2009, 57(3): 762-765.
- [9] Yusriski R, Samadhi T, Halim A. An integer batch scheduling model for a single machine with simultaneous learning and deterioration effects to minimize total actual flow time[C]. IOP Conf Series: Materials Science and Engineering. Kuala Lumpur: IOP Publishing, 2016, 114: 012073.
- [10] Li D C, Hsu P H, Chang C C. A genetic algorithm-based approach for single-machine scheduling with learning effect and release time[J]. Mathematical Problems in Engineering, 2014(1): 1-12.
- [11] Du J, Leung J Y T. Minimizing total tardiness on one machine is NP-hard[J]. Mathematics of Operations Research, 1990, 15(3): 483-495.
- [12] Hsu Y S, Lin B M T. Minimizing of maximum lateness under linear deterioration[J]. Omega-international J of Management Science, 2003, 31(6): 459-469.
- [13] Erena T, Güner E. Minimizing total tardiness in a scheduling problem with a learning effect[J]. Applied Mathematical Modelling, 2007, 31(7): 1351-1361.

作者简介

胡金昌(1992—),男,博士生,从事生产调度、物流系统仿真和优化的研究, E-mail: 201620367@mail.sdu.edu.cn;

吴耀华(1963—),男,教授,博士,从事自动拣选和仓储系统设计与优化等研究, E-mail: mike.wu@263.net;

吴颖颖(1985—),女,讲师,博士,从事物流系统仿真和优化的研究, E-mail: sophia.wu@sdu.edu.cn;

杨栋(1988—),男,博士生,从事物流系统规划与优化的研究, E-mail: yangdong.lv@foxmail.com.