

控制与决策

Control and Decision

单台批处理机生产运输协调合作博弈调度问题

宫华, 孙红梅, 孙文娟, 许可

引用本文:

宫华, 孙红梅, 孙文娟, 许可. 单台批处理机生产运输协调合作博弈调度问题[J]. *控制与决策*, 2024, 39(3): 956–964.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2022.1697>

您可能感兴趣的其他文章

Articles you may be interested in

[一种要素双模糊的限制交流结构合作博弈方法及应用](#)

An allocation model of limited communication structure cooperative game with dual fuzzy elements

控制与决策. 2021, 36(2): 475–482 <https://doi.org/10.13195/j.kzyjc.2019.1048>

[现货市场补充作用下基于总量折扣的运输服务采购问题研究](#)

Transportation service procurement based on total discount under complementary effect of spot market

控制与决策. 2021, 36(11): 2794–2802 <https://doi.org/10.13195/j.kzyjc.2020.0274>

[铁路集装箱中心站资源分配与作业调度联合优化](#)

Integrating optimization of resource allocation and handling scheduling in railway container terminal

控制与决策. 2021, 36(12): 3063–3073 <https://doi.org/10.13195/j.kzyjc.2020.0597>

[带峰值能耗约束流水线调度的协同群智能优化](#)

Cooperative memetic optimization for flowshop scheduling with peak power consumption constraint

控制与决策. 2021, 36(10): 2350–2358 <https://doi.org/10.13195/j.kzyjc.2020.0429>

[基于正态云模型的状态转移算法求解多目标柔性作业车间调度问题](#)

State transition algorithm based on normal cloud model for solving multi-objective flexible job shop scheduling problem

控制与决策. 2021, 36(5): 1181–1190 <https://doi.org/10.13195/j.kzyjc.2019.1233>

单台批处理机生产运输协调合作博弈调度问题

宫华^{1†}, 孙红梅², 孙文娟¹, 许可¹

(1. 沈阳理工大学 理学院, 沈阳 110159; 2. 沈阳理工大学 自动化与电气工程学院, 沈阳 110159)

摘要: 研究单台批处理机生产与生产前运输的协调调度问题, 目标函数为最小化与完成时间相关的生产总成本. 以工件为博弈方, 以联盟的最大成本节省为特征函数, 将调度问题转换为合作博弈模型. 针对相同运输时间与加工时间的情形, 证明该合作博弈具有非空核, β 规则可得一个核分配. 针对一般问题, 设计 Q -learning 算法求解联盟最优调度, 并利用 β 规则对节省的成本进行分配. 数值算例验证了合作博弈模型的可行性以及 Q -learning 算法与 β 规则对节省成本分配的有效性.

关键词: 批处理机; 运输; 合作博弈; Q -learning; 成本分配

中图分类号: TP278

文献标志码: A

DOI: 10.13195/j.kzyjc.2022.1697

引用格式: 宫华, 孙红梅, 孙文娟, 等. 单台批处理机生产运输协调合作博弈调度问题[J]. 控制与决策, 2024, 39(3): 956-964.

Production and transportation coordinated cooperative game scheduling problem of single batching machine

GONG Hua^{1†}, SUN Hong-mei², SUN Wen-juan¹, XU Ke¹

(1. School of Science, Shenyang Ligong University, Shenyang 110159, China; 2. School of Automation & Electrical Engineering, Shenyang Ligong University, Shenyang 110159, China)

Abstract: We study the coordinated scheduling problem of production and pre-production transportation for a single batching machine. The objective is to minimize the total production cost related to the completion time. The scheduling problem is transformed into a cooperative game model with the jobs as the players and the maximal cost savings of the coalition as the characteristic function. For the case of the same transportation time and processing time, it is proved that the cooperative game has a non-empty core, and the β rule can get a core allocation. For general problems, the Q -learning algorithm is designed to solve the optimal scheduling of the coalition, and the β rule is used to allocate the cost savings. Numerical examples verify the feasibility of the cooperative game model and the effectiveness of the Q -learning algorithm and β rule for cost savings allocation.

Keywords: batching machine; transportation; cooperative game; Q -learning; cost allocation

0 引言

生产物流系统是智能制造全流程的关键环节, 如何围绕多目标、多工序、多约束的生产运输全流程要素, 协同优化生产调度与物料运输, 是优化能源消耗、生产成本、生产效率的关键. 以钢铁企业为例, 模铸后的钢锭运输过程中温度降低, 在均热炉中成批加热, 以满足轧制温度要求. 均热炉是经典的批处理机调度问题, 生产、运输、客户之间存在冲突与约束, 不同主体中存在个体目标间的关联与冲突. 本文以模铸为背景, 以最小化工件总成本为目标, 提炼单台批

处理生产前运输协调调度问题.

针对带有批处理机生产运输协调调度问题的研究, 大多依靠传统调度理论和方法, 包括启发式算法^[1]、动态规划算法^[2-4]、智能优化算法等方法. 文献[5]考虑传搁时间对批处理机加工的影响, 研究生产前运输与生产协调调度问题. 文献[6]研究了批处理机生产前与生产后两阶段运输协调调度问题. 文献[7]研究多车运输的单台批处理机生产运输协调调度问题, 设计了 Q -learning 算法求解最优调度. 利用博弈理论研究带有批处理机生产运输协调调度问题的

收稿日期: 2022-09-26; 录用日期: 2022-12-01.

基金项目: 辽宁省“兴辽英才计划”项目(XLYC2006017); 辽宁省教育厅科学研究经费项目(LG202025, LJKQZ 2021057).

责任编辑: 王凌.

[†]通讯作者. E-mail: gonghua@sylu.edu.cn.

相关研究较少,文献[8]研究了带尺寸约束的二机流水车间生产运输调度问题,设计 Q -learning算法求解纳什均衡调度.对批处理机生产运输协调调度问题的研究,大多从企业角度考虑整体利益使目标达到最优,忽略了不同客户之间可通过合作结盟的方式重新安排生产顺序,得到令客户满意的调度方案.

合作博弈理论应用到生产调度问题中大多集中在单机、并行机^[9-10]与流水车间环境.文献[11]最早研究具有线性成本系数的单机排序博弈问题,证明了合作博弈是平衡博弈,给出了EGS规则分配成本节省.文献[12]在文献[11]排序博弈研究基础上考虑了工件位置和学习效应.文献[13]研究了工件加工时间相同,准备时间为加工时间倍数的单机博弈调度问题.文献[14-16]分别研究了带有工件恶化、带有工件交货期、工件具有外部性的单机合作博弈调度问题.文献[17]研究了工件具有紧急度指数单机调度合作博弈问题,具有非空核.文献[18]研究了比例流水车间调度合作博弈问题,基于提前及拖期惩罚的 β 规则分配节省成本.文献[19]研究了单台批处理机和由批处理机组成的流水车间的调度合作博弈(是凸博弈),基于Shapley值、EGS规则分配成本节省.综上,针对不同车间环境博弈调度问题的研究,主要集中在生产阶段,较少考虑生产与运输的协调.

本文应用合作博弈理论研究单台批处理机生产与运输协调调度问题,以工件为博弈方建立合作博弈调度模型.博弈方基于个体理性和集体理性原则,通过结盟的方式使所有工件的总成本节省最大.针对具有相同运输时间及相同加工时间这类特殊问题,分析其博弈性质;对于一般问题,通过强化学习设计 Q -learning算法求解联盟最优调度顺序,得到各联盟的成本节省,并对节省的成本进行分配.

1 问题描述

问题描述如下:工件集合 N 包含 n 个客户,每个客户只有一个工件(或订单),工件的加工成本为工件在机器上完工时间的线性函数.批处理机可同时加工多个工件,每个工件只能被加工一次.工件具有初始调度顺序,一台运输车将仓库中工件运输到批处理机前等待加工,一次只能运输一个工件,考虑工件的运输时间,运输车运输完一个工件后返回仓库开始下一次运输.假设:1)所有工件0时刻在仓库等待运输;2)批处理机前有无限缓冲区;3)批处理机开始加工后不允许中断;4)运输车从开始运输到完成无空闲时间.

相关符号及说明如下:

p_j : 工件 j 的加工时间;
 t_j : 工件 j 的运输时间;
 ω_j : 工件 j 的成本系数;
 C_j : 工件 j 的完工时间;
 T : 运输车空车返回时间;
 trans_j : 工件 j 的运输完成时间;
 c : 批处理机容量;
 b : 工件总批数, $\lceil n/c \rceil \leq b \leq n$;
 B_i : 批处理机中加工的第 i 批工件集合;
 S_i : 第 i 批工件的开始加工时间;
 P_i : 第 i 批工件的加工时间;
 A_i : 第 i 批工件的完工时间.

数学模型如下:

$$\min Z = \sum_{j=1}^n \omega_j C_j. \quad (1)$$

$$\text{s.t.} \sum_{i=1}^b x_{j,i} = 1, j = 1, 2, \dots, n. \quad (2)$$

$$\sum_{j=1}^n x_{j,i} \leq c, i = 1, 2, \dots, b. \quad (3)$$

$$P_i = \max_{j \in B_i} \{p_j\}, i = 1, 2, \dots, b. \quad (4)$$

$$\sum_{i=1}^b \sum_{j=1}^n x_{j,i} = n. \quad (5)$$

$$S_1 = \sum_{j=1}^n (t_j + T)x_{j,1} - T. \quad (6)$$

$$S_i = \max \left\{ A_{i-1}, \sum_{k=1}^i \sum_{j=1}^n (t_j + T)x_{j,k} - T \right\}, i > 1. \quad (7)$$

$$A_i = S_i + P_i. \quad (8)$$

$$C_j = \sum_{i=1}^b A_i x_{j,i}. \quad (9)$$

$$x_{j,i} = \begin{cases} 1, & \text{若工件} j \text{在第} i \text{批中加工;} \\ 0, & \text{otherwise;} \end{cases} \quad (10)$$

其中:式(2)表示每个工件只能加工一次;式(3)表示批处理机容量限制;式(4)表示每批工件的加工时间;式(5)表示所有工件被加工完成;式(6)表示第1批工件的开始加工时间;式(7)表示第 i ($i > 1$)批工件的开始加工时间;式(8)和(9)分别表示第 i 批工件的完工时间和工件 j 的完工时间;式(10)为决策变量.

2 合作博弈模型

单台批处理机生产与运输协调调度问题可用三元组 (N, σ_0, ω) 表示.其中: N 表示工件集合; σ_0 为工

件初始调度顺序, $\sigma_0(j) = k$ 表示工件 j 处于调度顺序 σ_0 中第 k 个位置; $\omega = (\omega_j)_{j \in N}$ 为工件成本系数集合. 当工件的加工时间为确定值 P_M 时, 问题用四元组 $(N, \sigma_0, \omega, P_M)$ 表示.

定义1 合作博弈的特征型用有序数对 (N, v) 表示, 其中 v 是 N 所有子集上的映射, 即 $v : 2^N \rightarrow \mathbb{R}$ 且 $v(\emptyset) = 0$, 则 v 是联盟对应的特征函数, 联盟 $S (S \subseteq N)$ 的特征值记作 $v(S)^{[11]}$.

定义2 对于联盟 $S (S \subseteq N)$, 若 $\forall i, j \in S, k \in N$, 当 $\sigma_0(i) < \sigma_0(k) < \sigma_0(j)$ 时, $k \in S$ 成立, 则称联盟 S 关于调度顺序 σ_0 是连通的^[11].

定义3 对于联盟 $S (S \subseteq N)$ 内联通联盟 T , 若 $\forall j \in \{S \setminus T\}, T \cup \{j\}$ 关于调度顺序 σ_0 均不连通, 则称 T 为联盟 S 的一个 σ_0 -组^[11], 联盟 S 的 σ_0 -组集合记为 S/σ_0 .

定义4 若 n 维向量 $x = (x_1, x_2, \dots, x_n)$ 满足 $\sum_{j \in N} x_j = v(N)$, 且对于任意联盟 $S (S \subseteq N)$ 均满足 $\sum_{j \in S} x_j \geq v(S)$, 则称 x 为一个核分配, 满足这个条件配置的集合称为合作博弈的核^[11], 记作 $C(v)$.

$$C(v) = \left\{ x \in \mathbb{R}^n \mid \sum_{j \in N} x_j = v(N), \sum_{j \in S} x_j \geq v(S), \forall S \in 2^N \right\}. \quad (11)$$

显然, 若 x 为核分配, 则满足集体理性, 能够将大联盟带来的成本节省全部分配完毕, 而且满足个体理性, 即 $x_j \geq v(\{j\})$, 合作带来的成本节省分配总是大于单干得到的值. 此时, 在核分配下, 大联盟能够稳定存在, 没有博弈方愿意偏离.

定义5 若合作博弈 (N, v) 满足: 对于任意 $j \in N, v(\{j\}) = 0$; (N, v) 满足超可加性; 对 $S (S \subseteq N)$ 有 $v(S) = \sum_{T \in S/\sigma_0} v(T)$. 则称合作博弈 (N, v) 是 σ_0 -组可加博弈^[11].

以工件为博弈方, 以联盟内工件重调度获得的最大成本节省为特征函数, 建立合作博弈模型. 问题中要求 σ_0 -组内工件方可交换调度顺序, 且不能损害联盟外工件的利益, 将联盟 S 的可行调度集记作 Π_S . 联盟 S 的特征值为初始调度总成本与联盟最优调度总成本的差值, 即

$$v(S) = \max_{\sigma \in \Pi_S} \sum_{j \in N} \omega_j [C_j(\sigma_0) - C_j(\sigma)]. \quad (12)$$

3 $(N, \sigma_0, \omega, P_M)$ 问题的合作博弈

当工件运输时间、加工时间均为常数, 即 $t_j = T, p_j = P_M, j \in N$ 时, 确定工件调度顺序后, 还需

确定工件如何分批. 工件的总批次 b 的取值范围为 $(\lceil n/c \rceil \leq b \leq n)$. LOE 规则是指前 $(\lceil n/c \rceil - 1)$ 批包含 c 个工件, 均为满批, 最后一批包含 $[n - (\lceil n/c \rceil - 1)c]$ 个工件. 本文从生产资源利用率角度出发, 根据 LOE 规则对工件分批, 可以降低机器负载.

若调度顺序为 σ , 则工件 j 的运输完成时间 trans_j 、完工时间 C_j 如下所示:

$$\text{trans}_j = (2k - 1)T, \sigma(j) = k, k \in N; \quad (13)$$

$$C_j = \text{trans}_j + P_M \lceil \sigma(j)/c \rceil. \quad (14)$$

联盟内工件需要重新安排调度顺序使总成本最小. $(N, \sigma_0, \omega, P_M)$ 问题的最优调度由 HCFF (highest cost factor first) 规则确定, 工件按照成本系数非增序规则排序. 这是因为目标函数由工件完工时间与成本系数共同确定, 而完工时间取决于工件在调度顺序中位置, 因此工件按成本系数非增序排序, 得到的即是工件的最优调度.

3.1 合作博弈性质分析

针对单台批处理机生产与运输协调调度问题 $(N, \sigma_0, \omega, P_M)$, 工件的成本仅与其在调度顺序中的位置有关, 联盟 S 内工件交换顺序不影响联盟外工件的成本, 因此合作博弈无外部性.

性质1 $(N, \sigma_0, \omega, P_M)$ 的合作博弈 (N, v) 具有超可加性, 即对于任意联盟 $S, T \subset N, S \cap T = \emptyset$, 有 $v(S) + v(T) \leq v(S \cup T)$.

证明 假设联盟 $S, T \subset N, S \cap T = \emptyset$, 联盟 S, T 的最优调度顺序分别为 σ_S, σ_T , 其中 σ_S 或 σ_T 为联盟 S 或 T 外工件调度顺序与 σ_0 相同, 联盟 S 或 T 内工件按照工件成本系数非增序排列 (HCFF 规则) 得到的最优调度顺序. 可得

$$\begin{aligned} v(S) + v(T) &= \sum_{j \in S} \omega_j [C_j(\sigma_0) - C_j(\sigma_S)] + \sum_{j \in T} \omega_j [C_j(\sigma_0) - C_j(\sigma_T)] = \sum_{j \in (S \cup T)} \omega_j [C_j(\sigma_0) - C_j(\sigma'_{(S \cup T)})] \leq \max_{\sigma \in \Pi_{(S \cup T)}} \sum_{j \in (S \cup T)} \omega_j [C_j(\sigma_0) - C_j(\sigma)] = v(S \cup T). \end{aligned}$$

其中: $\sigma'_{(S \cup T)}$ 表示联盟 S 及 T 内工件调度顺序分别与 σ_S 及 σ_T 一致, 联盟 S 及联盟 T 外工件的调度顺序与初始调度 σ_0 相同; $\Pi_{(S \cup T)}$ 表示联盟 S 及联盟 T 的并集构成的联盟的所有可行调度. 显然 $\sigma'_{(S \cup T)} \in \Pi_{(S \cup T)}$, 因此, 合作博弈具有超可加性. \square

性质2 $(N, \sigma_0, \omega, P_M)$ 的合作博弈 (N, v) 是 σ_0 -组可加博弈.

证明 1) $\forall j \in N$, 显然满足 $v(\{j\}) = 0$;

2) 由性质1可知, 合作博弈 (N, v) 满足超可加性;

3) 对于 $\forall S(S \subseteq N), T \in \{T_1, T_2, \dots, T_{|S/\sigma_0|}\} = S/\sigma_0$, 由于合作博弈具有无外部性, 有

$$\begin{aligned} v(S) &= \max_{\sigma \in \Pi_S} \sum_{j \in N} \omega_j [C_j(\sigma_0) - C_j(\sigma)] = \\ &= \max_{\sigma \in \Pi_S} \sum_{j \in S} \omega_j [C_j(\sigma_0) - C_j(\sigma)] = \\ &= \max_{\sigma_1 \in \Pi_{T_1}} \sum_{j \in T_1} \omega_j [C_j(\sigma_0) - C_j(\sigma_1)] + \dots + \\ &= \max_{\sigma_{|S/\sigma_0|} \in \Pi_{T_{|S/\sigma_0|}}} \sum_{j \in T_{|S/\sigma_0|}} \omega_j [C_j(\sigma_0) - C_j(\sigma_{|S/\sigma_0|})] = \\ &= v(T_1) + \dots + v(T_{|S/\sigma_0|}) = \sum_{T \in S/\sigma_0} v(T). \end{aligned}$$

即 $v(S) = \sum_{T \in S/\sigma_0} v(T)$, 合作博弈 (N, v) 是 σ_0 -组可加博弈, 从而是平衡博弈, 具有非空核. \square

3.2 成本分配

工件结盟后通过交换加工顺序可节省成本, 需对结盟节省的成本进行合理分配, 保证联盟稳定存在. 本文采用 β 规则对节省成本进行分配, 定义以下符号:

$F(\sigma, j) = \{k \in N | \sigma(k) < \sigma(j)\}$: 排在工件 j 前面的工件集合;

$\bar{F}(\sigma, j) = \{k \in N | \sigma(k) \leq \sigma(j)\}$: 工件 j 及排在工件 j 前的工件集合;

$L(\sigma, j) = \{k \in N | \sigma(k) > \sigma(j)\}$: 排在工件 j 后的工件集合;

$\bar{L}(\sigma, j) = \{k \in N | \sigma(k) \geq \sigma(j)\}$: 工件 j 及排在工件 j 后的工件集合.

β 规则按工件加入其前后工件集产生的边际成本加权求和进行分配, 计算方式如下:

$$\begin{aligned} x_j(v) &= \beta[v(\bar{F}(\sigma, j)) - v(F(\sigma, j))] + \\ &= (1 - \beta)[v(\bar{L}(\sigma, j)) - v(L(\sigma, j))]. \end{aligned} \quad (15)$$

性质3 基于 β 规则得到的成本分配是 $(N, \sigma_0, \omega, P_M)$ 问题合作博弈的一个核分配.

证明 假设工件初始顺序为 $\sigma_0 = (1, 2, \dots, n)$, 有 $\bar{F}(\sigma_0, n) = N = \bar{L}(\sigma_0, 1), F(\sigma_0, j) = \bar{F}(\sigma_0, j - 1), L(\sigma_0, j - 1) = \bar{L}(\sigma_0, j)$. 大联盟 N 中所有工件成本分配之和为

$$\sum_{j \in N} x_j(v) =$$

$$\begin{aligned} &= \beta[v(\bar{F}(\sigma_0, 1)) - v(F(\sigma_0, 1))] + \\ &= (1 - \beta)[v(\bar{L}(\sigma_0, 1)) - v(L(\sigma_0, 1))] + \dots + \\ &= \beta[v(\bar{F}(\sigma_0, n)) - v(F(\sigma_0, n))] + \\ &= (1 - \beta)[v(\bar{L}(\sigma_0, n)) - v(L(\sigma_0, n))] = \\ &= \beta v(\bar{F}(\sigma_0, n)) + (1 - \beta)v(\bar{L}(\sigma_0, 1)) = \\ &= v(N). \end{aligned}$$

即 $\sum_{j \in N} x_j(v) = v(N)$, 大联盟节省成本 $v(N)$ 全部分配给联盟内工件.

对于 $\forall S \subseteq N, T \in S/\sigma_0$, 设 f 和 l 分别为联盟 T 中在初始顺序下第一个和最后一个工件序号. 参与联盟的工件分配到的收益之和为

$$\begin{aligned} \sum_{j \in S} x_j &= \sum_{T \in S/\sigma_0} \sum_{j \in T} \{\beta[v(\bar{F}(\sigma_0, j)) - v(F(\sigma_0, j))] + \\ &= (1 - \beta)[v(\bar{L}(\sigma_0, j)) - v(L(\sigma_0, j))]\} = \\ &= \sum_{T \in S/\sigma_0} \{\beta[v(\bar{F}(\sigma_0, l)) - v(F(\sigma_0, f))] + \\ &= (1 - \beta)[v(\bar{L}(\sigma_0, f)) - v(L(\sigma_0, l))]\}. \end{aligned}$$

又因

$$\begin{aligned} v(\bar{F}(\sigma_0, l)) - v(F(\sigma_0, f)) &\geq v(T), \\ v(\bar{L}(\sigma_0, f)) - v(L(\sigma_0, l)) &\geq v(T), \end{aligned}$$

所以

$$\begin{aligned} \sum_{j \in S} x_j &\geq \sum_{T \in S/\sigma_0} \{\beta v(T) + (1 - \beta)v(T)\} = \\ &= \sum_{T \in S/\sigma_0} v(T), \\ v(S) &= \sum_{T \in S/\sigma_0} v(T) \Rightarrow \sum_{j \in S} x_j \geq v(S). \end{aligned}$$

综上, 基于 β 规则的成本分配是 $(N, \sigma_0, \omega, P_M)$ 问题合作博弈的核分配. \square

3.3 算例分析

为验证单台批处理机调度的合作博弈模型的性质, 根据 β 规则对节省成本进行分配 ($\beta = 0.5$). 设 $n = 5, c = 2, T = 1, P_M = 1$. 工件的初始调度顺序为 $\sigma_0 = (1, 2, 3, 4, 5)$, 成本系数为 $\omega = (2, 5, 6, 7, 8)$, 批处理机最优调度顺序由 HCFF 规则确定, 初始排序下工件成本 $C = (8, 20, 48, 56, 80)$, 所有工件的总成本为 212. 根据可行调度定义与 HCFF、LOE 规则, 得到各连通联盟的最大成本节省. 各连通联盟的成本节省如表 1 所示.

表1 各连通联盟的成本节省

联盟 S	成本节省 $v(S)$	联盟 S	成本节省 $v(S)$
1, 2	0	2, 3, 4	8
2, 3	4	3, 4, 5	4
3, 4	0	1, 2, 3, 4	24
4, 5	2	2, 3, 4, 5	18
1, 2, 3	16	1, 2, 3, 4, 5	44

大联盟 N 能够获得最大成本节省为 44, 利用 β 规则对各工件分配的成本节省值为 $x_j(v) = (13, 7, 9, 5, 10)$, 显然 $\sum_{j \in N} x_j(v) = 44$, 且对于每个连通联盟均有 $\sum_{j \in S} x_j(v) \geq v(S)$, 说明按照 β 规则分配成本节省是核分配.

4 (N, σ_0, ω) 问题的合作博弈

对于单台批处理机生产前运输协调调度问题的一般情况 (N, σ_0, ω), 工件的成本由工件的运输时间、批处理机加工时间和成本系数决定, 设计 Q -learning 算法求解使联盟获得最大成本节省的调度方案.

4.1 问题转化

Q -learning 算法求解调度问题时需将调度问题转化为一个半马尔科夫问题.

4.1.1 状态特征

状态描述系统的整体特征和局部特征, 定义运输车、工件和批处理机的状态变化描述整个系统. 触发状态发生转移的事件包括新的工件运输到批处理机前、运输车空车返回仓库和批处理机完成一批工件加工. 为满足不同问题规模要求, 状态采用归一化形式表示. 本节共定义 $n + 10$ 种状态特征, 其中运输车具有 4 个状态特征 ($f_{1,k}$ 表示运输车的第 k 个状态); 批处理机具有 6 个状态特征 ($f_{2,k}$ 表示批处理机的第 k 个状态); $f_{j+2,1} (1 \leq j \leq n)$ 表示各阶段工件状态. Q_V 表示运输车前等待运输的工件序列, $|Q_V|$ 表示运输车前等待序列中工件数量; Q_M 表示批处理机前等待加工工件序列, $|Q_M|$ 表示等待加工工件序列中工件数量, 状态特征信息见表 2.

表2 状态特征信息表

类别	序号	状态特征函数	描述
运输车	1	$f_{1,1} = \begin{cases} j/n, & \text{运输车正在运输工件 } j; \\ 0, & \text{运输车返回仓库运输下一个工件;} \\ -1, & \text{运输车空闲} \end{cases}$	运输车运输工件情况
	2	$f_{1,2} = \frac{ Q_V }{n}$	运输车前等待运输工件数量
	3	$f_{1,3} = \begin{cases} 0, & \arg \max_{j \in N} t_j \text{ 的工件不在等待队列 } Q_V \text{ 中;} \\ 1, & \arg \max_{j \in N} t_j \text{ 的工件在等待队列 } Q_V \text{ 中} \end{cases}$	运输时间最长的工件是否在运输车前等待队列中
	4	$f_{1,4} = \begin{cases} 0, & \arg \min_{j \in N} t_j \text{ 的工件不在等待队列 } Q_V \text{ 中;} \\ 1, & \arg \min_{j \in N} t_j \text{ 的工件在等待队列 } Q_V \text{ 中} \end{cases}$	运输时间最短的工件是否在运输车前等待队列中
批处理机	5	$f_{2,1} = \begin{cases} 0, & \text{批处理机空闲;} \\ 1/i, & \text{批处理机正在加工第 } i \text{ 批工件} \end{cases}$	批处理机加工工件批次信息
	6	$f_{2,2} = \frac{ Q_M }{n}$	批处理机前等待加工工件数量
	7	$f_{2,3} = \frac{t - S_i}{\max_{j \in N} \{p_j\}}$	t 时刻, 批处理机加工第 i 批工件的剩余加工时间与所有工件加工时间最大值之比
	8	$f_{2,4} = \frac{\max_{j \in Q_M} \{p_j\}}{\max_{j \in N} \{p_j\}}$	等待加工序列中加工时间最大值与所有工件加工时间最大值之比
	9	$f_{2,5} = \frac{\min_{j \in Q_M} \{p_j\}}{\max_{j \in N} \{p_j\}}$	等待加工序列中加工时间最小值与所有工件加工时间最大值之比
	10	$f_{2,6} = \frac{\sum_{j \in Q_M} p_j}{ Q_M } \cdot \frac{n}{\sum_{j=1}^n p_j}$	等待加工序列中工件的平均加工时间与所有工件的平均加工时间之比
工件	11	$f_{j+2,1} = \begin{cases} 0, & \text{工件 } j \text{ 等待运输;} \\ 1, & \text{工件 } j \text{ 正在被运输;} \\ -1, & \text{工件 } j \text{ 在批处理机前缓冲区等待加工;} \\ 1/2, & \text{工件 } j \text{ 正在批处理机加工;} \\ -1/2, & \text{工件 } j \text{ 加工完成} \end{cases}$	工件状态

4.1.2 动作

动作定义与状态定义有关,反映在当前状态下系统可选动作的集合.批处理机生产与运输协调调度问题可将智能体动作分为运输车动作和批处理机动作.智能体根据运输车当前状态,选择优先运输的工件.运输车动作定义如下:

动作1(V1):最短运输时间优先规则,等待运输工件按运输时间非降序排列,优先运输时间短的工件.

动作2(V2):最长运输时间优先规则,等待运输工件按运输时间非增序排列,优先运输时间长的工件.

动作3(V3):最大成本系数优先规则,等待运输工件按成本系数非增序排列,优先运输成本系数大的工件.

动作4(V4):当运输车正在运输工件、运输车返回或仓库中所有工件都已运输完成,不选择工件运输.

批处理机动作的目的是从机器前等待加工的工件集合 Q_M 中,选出 J 个工件进行加工,其中 $J = \min\{c, |Q_M|\}$.批处理机动作定义如下:

动作5(FCFS):按照运输车运输工件到达批处理机前的顺序选择 J 个工件.

动作6(SPT):工件按加工时间非降序排列,选择 J 个工件放入批中.

动作7(LPT):工件按加工时间非增序排列,选择 J 个工件放入批中.

动作8(FB):按照工件的到达顺序,选择 c 个工件放入批中,构成满批.

动作9(select no job):批处理机正在加工工件或所有工件均已加工完成,选择该动作.

4.1.3 奖励函数

奖励函数的定义与单台批处理机生产前运输协调调度问题优化目标-最小化工件总成本相关.即时奖励反映智能体在当前状态选择动作的即时效果,累积奖励反映选择动作的长期效果,表示工件总成本.工件状态的示性函数定义为

$$\delta_j(t) = \begin{cases} 0, & \text{在时刻}t\text{工件}j\text{加工完成;} \\ -1, & \text{otherwise.} \end{cases} \quad (16)$$

奖励函数定义为

$$r_k = \sum_{j=1}^n \int_{t_{k-1}}^{t_k} \omega_j \delta_j(t) dt. \quad (17)$$

r_k 为系统在时刻 t_{k-1} 执行动作后,发生一次状态转移获得的奖励.最小化所有工件总成本等价于

一次迭代的最大累积奖励 R ,满足

$$\begin{aligned} R &= \sum_{k=1}^K r_k = \sum_{k=1}^K \sum_{j=1}^n \int_{t_{k-1}}^{t_k} \omega_j \delta_j(t) dt = \\ &= \sum_{j=1}^n \sum_{k=1}^K \int_{t_{k-1}}^{t_k} \omega_j \delta_j(t) dt = \\ &= \sum_{j=1}^n \int_0^{C_j} \omega_j \delta_j(t) dt = \\ &= \sum_{j=1}^n (-\omega_j C_j). \end{aligned}$$

由上可知,工件的总成本越小获得的累积奖励越大,最小化工件总成本等价于最大化累积奖励.即

$$\min \sum_{j=1}^n \omega_j C_j \Leftrightarrow \max \sum_{k=1}^K \sum_{j=1}^n \int_{t_{k-1}}^{t_k} \omega_j \delta_j(t) dt.$$

4.2 Q-learning 算法求解联盟最优调度

4.2.1 探索与利用

采用 ϵ 贪婪策略选择动作, $Q(s, a)$ 为状态 s 下选择动作 a 的值函数,每次迭代选择当前状态下使 $Q(s, a)$ 值最大的动作.以 ϵ 的概率探索环境信息,随机选择动作;以 $1 - \epsilon$ 的概率利用环境信息,在当前状态下选择使 $Q(s, a)$ 最大的动作. ϵ 贪婪策略的数学表达式为

$$A(s) = \begin{cases} \text{Random Action, } P = \epsilon; \\ \arg \max_a Q(s, a), P = 1 - \epsilon. \end{cases} \quad (18)$$

4.2.2 算法框架

采用函数逼近的Q-learning算法求解可行调度下各联盟的成本节省,通过更新基函数权重更新 $Q(s, a)$,更新公式为

$$Q(s, a) = \sum_{k=1}^{n+10} \eta_k^a \Phi_k(s). \quad (19)$$

其中: $n + 10$ 为状态向量中分量个数, $\Phi_k(s)$ 表示基函数, η_k^a 表示在当前状态下选择动作 a 的权重,权重向量由梯度下降法进行更新,正规化基函数为

$$\Phi_k(s) = \begin{cases} f_{1,k}, & 1 \leq k \leq 4; \\ f_{2,k-4}, & 5 \leq k \leq 10; \\ f_{k-8,1}, & 11 \leq k \leq n + 10. \end{cases} \quad (20)$$

Q-learning算法步骤如下:

step 1: 初始化问题参数和Q-learning算法参数.

问题参数:工件数量、批处理机容量、工件加工时间、运输时间、成本系数、运输车空车返回时间;算法参数:学习率 α 、衰减率 λ 、折扣因子 γ 、贪婪因子 ϵ 、基函数的权重 $\eta^a = (1, 1, \dots, 1)_{n+10}$ 、资格迹 $E(a) = (0, 0, \dots, 0)_{n+10}$.

- step 2: 定义初始时刻、状态.
- step 3: 根据 ε 贪婪策略选择联合行为并执行.
- step 4: 计算状态转移时间,更新当前时刻,计算奖励,更新状态.
- step 5: 更新基函数权重,更新过程如下所示:

$$E(a_k) = \lambda E(a_k) + \nabla_{\eta_k^a} Q(s_k, a_k),$$

$$\delta(a_k) = r(s_k, a_k, s_{k+1}) - Q(s_k, a_k) + \gamma \max_{a_{k+1} \in A(s_{k+1})} Q(s_{k+1}, a_{k+1}),$$

$$\eta_k^a = \eta_k^a + \alpha \delta(a_k) E(a_k).$$

- step 6: 判断批处理机是否加工完成全部工件,若小于工件数量,则执行 step 3, 否则,执行 step 7;
- step 7: 判断当前迭代次数是否小于最大迭代次数,若小于,则迭代次数加 1, 执行 step 2; 否则,输出工件最优调度顺序, 结束.

算法流程如图 1 所示.

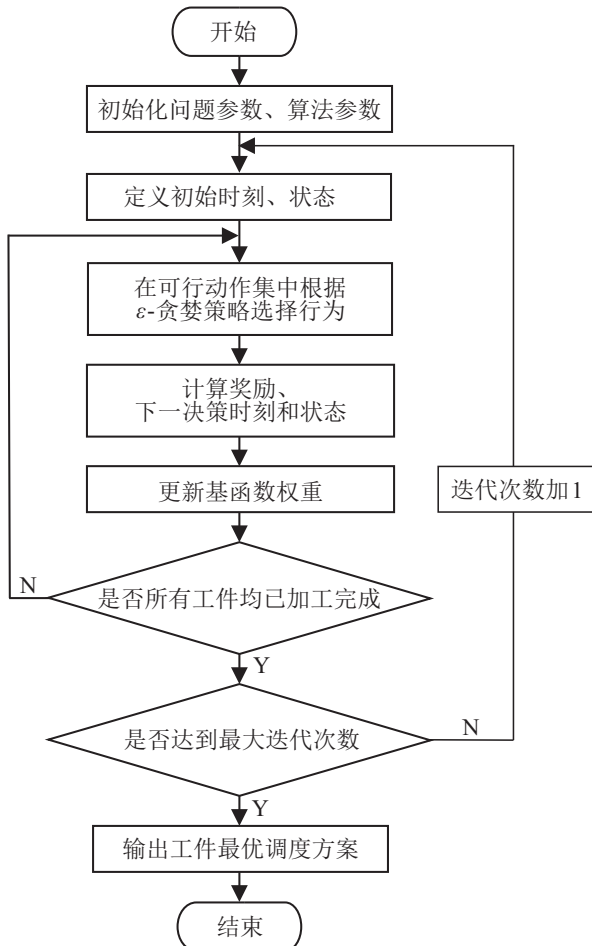


图 1 Q-learning 算法流程

4.3 实验仿真

4.3.1 实验环境及参数设置

本文通过 Python 语言在 JetBrains PyCharm Community Edition 2018.3.3 软件上编写程序, 计算机安装内存为 4.00 GB, 处理器为 Intel (R) Core (TM) i5-

6200U CPU @2.30 GHz.

参数设置如下: 工件的运输时间 $t_j \sim U[5, 15]$, 加工时间 $t_j \sim U[1, 50]$, 成本系数 $\omega_j \sim U[1, 5]$, 运输车返回时间 $T \sim U[1, 10]$, 批处理机容量 $c \sim \text{random}[1, 10]$. 利用三水平四因素正交实验法确定实验中 Q-learning 算法参数, $\alpha, \lambda, \gamma, \varepsilon$ 参数初始水平为 (0.01, 0.08, 0.1, 0.1), (0.02, 0.1, 0.3, 0.15), (0.03, 0.15, 0.5, 0.2). 实验中选择问题参数 $n = 15, c = 2, T = 3$, 最大迭代次数为 500 次, 按照三水平四因素 (L_9) 规则对各算法参数进行交换, 确定算法参数取值为: $\alpha = 0.02, \lambda = 0.08, \gamma = 0.5, \varepsilon = 0.15$.

4.3.2 不同规模问题的 Q-learning 算法实验结果

为验证 Q-learning 算法求解联盟最优调度的稳定性, 选择不同问题规模进行实验. 参数 $c = 2, T = 3$, 工件数量分别取 15、30、40、50、70、100, 最大迭代次数为 500 次. 启发式规则由运输车规则-批处理机规则两部分组成, 对比不同启发式规则与 Q-learning 算法得到的工件总成本, 结果如表 3 所示.

表 3 启发式规则与 Q-learning 算法得到的工件总成本

规则	总成本					
	$n = 15$	$n = 30$	$n = 40$	$n = 50$	$n = 70$	$n = 100$
V1-FCFS	6752	18881	32356	51631	93401	181019
V1-SPT	6690	18897	32328	51542	93396	181505
V1-LPT	6812	18881	32356	52879	93402	180998
V1-FB	6752	18881	32356	51631	93401	181019
V2-FCFS	12914	37985	63349	94169	191987	368470
V2-SPT	12517	36645	61623	91733	187008	364131
V2-LPT	12914	37641	62932	93434	190614	366851
V2-FB	12514	37985	63349	94169	191987	368470
V3-FCFS	12921	37872	63630	91851	189285	356941
V3-SPT	12517	36645	61623	91617	186892	359659
V3-LPT	12914	37641	62932	89983	185597	348943
V3-FB	12921	37872	63630	91851	189285	356941
Q-learning	6234	15563	27912	38164	75842	138438

由表 3 可知, 针对不同问题规模, Q-learning 算法与其他启发式算法相比成本值最少降低了 6.8%, 最多降低了 62.42%, 相比于启发式算法的平均成本值降低了 42.07%. 实验表明: 与启发式算法相比, Q-learning 算法可得到联盟内工件的最优调度顺序, 使工件的总成本最小, 从而得到最大成本节省.

4.3.3 基于合作博弈求解调度问题的结果分析

在初始调度下不考虑工件之间的合作结盟, 运输车、批处理机空闲, 工件即可被运输、加工. 选取工件数量 $n = 15$, 问题参数 $c = 2, T = 3$, 按照客户提供订单顺序确定工件在运输车上的初始调度顺序 $\sigma_0 = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15)$, 运输

车按照初始顺序 σ_0 运输工件,批处理机空闲即可加工运输完成的工件,若等待运输区工件数量小于批容量,则工件不会等待其他工件到达后以满批形式加工.初始调度下工件总成本为7495.当工件合作结成联盟时,利用 Q -learning 算法求解联盟的最优调度方案,得到各连通联盟的成本节省值.大联盟最优调度顺序下的工件总成本为6234,最优调度甘特图如图2所示.

联盟 σ_0 -组内的工件可任意交换调度顺序,联盟外工件的调度顺序保持不变,通过合作结盟方式使集体利益达到最大. Q -learning 算法得到各连通联盟的

成本节省如表4所示,其中联盟 $\{a - b\}$ 表示从 a 到 b 的连通集,如联盟 $\{3-6\}$ 表示连通联盟 $\{3, 4, 5, 6\}$.

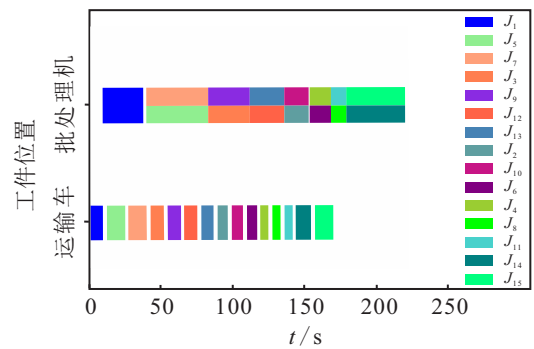


图2 工件最优调度甘特图

表4 各连通联盟的成本节省

联盟 S	成本节省 $v(S)$	联盟 S	成本节省 $v(S)$	联盟 S	成本节省 $v(S)$	联盟 S	成本节省 $v(S)$	联盟 S	成本节省 $v(S)$
1,2	265	8-10	148	4-8	894	4-10	894	1-10	1180
2,3	0	9-11	146	5-9	894	5-11	894	2-11	1258
3,4	422	10-12	0	6-10	297	6-12	357	3-12	1108
4,5	0	11-13	0	7-11	297	7-13	336	4-13	1143
5,6	894	12-14	0	8-12	162	8-14	281	5-14	1068
6,7	0	13-15	0	9-13	186	9-15	186	6-15	505
7,8	297	1-4	422	10-14	39	1-8	894	1-11	1258
8,9	0	2-5	422	11-15	0	2-9	1180	2-12	1258
9,10	7	3-6	894	1-6	894	3-10	894	3-13	1258
10,11	0	4-7	894	2-7	894	4-11	1077	4-14	1143
11,12	0	5-8	894	3-8	894	5-12	1068	5-15	1068
12,13	0	6-9	297	4-9	894	6-13	387	1-12	1258
13,14	0	7-10	297	5-10	894	7-14	429	2-13	1261
14,15	0	8-11	148	6-11	357	8-15	281	3-14	1258
1-3	265	9-12	146	7-12	297	1-9	1180	4-15	1143
2-4	422	10-13	39	8-13	208	2-10	1180	1-13	1261
3-5	422	11-14	0	9-14	186	3-11	1108	2-14	1261
4-6	894	12-15	0	10-15	39	4-12	1082	3-15	1258
5-7	894	1-5	422	1-7	894	5-13	1068	1-14	1261
6-8	297	2-6	894	2-8	894	6-14	429	2-15	1261
7-9	297	3-7	894	3-9	894	7-15	429	1-15	1261

由表4可知 $v(N) = 1261$,利用 β 规则对节省成本进行分配,各工件分配值 $x(v) = (0, 134, 57.5, 116, 281.5, 274, 74, 47.5, 216.5, 19.5, 39, 0, 1.5, 0, 0)$. 显然 $\sum_{j \in N} x_j(v) = v(N) = 1261$ 且 $x_j \geq v(\{j\}) = 0$, 表明利用 β 规则得到的成本分配是一种有效分配,满足集体理性和个体理性. 对联盟 $\{3, 4\}$ 有 $\sum_{j \in \{3,4\}} x_j = 173.5 < 422$,表明在一般问题中, β 规则得到的成本分配并不是一种核分配. 由于 $v(\{2-13\}) = v(N)$,客户 1、14、15 对大联盟没有贡献,成本分配值为0;客户 5、6、9 参与的联盟均获得较大成本节省,因此分配到较多的成本节省值.

5 结论

本文研究单台批处理机生产与生产前运输协调调度问题. 以工件为博弈方,联盟内仅 σ_0 -组中的工件可交换调度顺序,以节省成本为联盟的特征函数,建立合作博弈调度模型. 针对 $(N, \sigma_0, \omega, P_M)$ 问题,证明该合作博弈无外部性且是 σ_0 -组可加博弈,从而具有非空核. HCCFF 规则可确定各联盟工件最优调度顺序. 为保证大联盟稳定存在,采用 β 规则对节省成本进行分配并证明其是一种核分配. 通过算例验证了合作博弈模型及成本分配方法的有效性. 针对一般问题 (N, σ_0, ω) ,设计 Q -learning 算法求解各联盟的最优调度,在不损害联盟外工件利益基础上,得到各连

通联盟的特征值,并利用 β 规则对节省成本进行分配.实验结果验证了 Q -learning算法求解最优调度以及成本分配方式的可行性和有效性.

未来将会继续对多台运输车及多台批处理机等具有更复杂生产运输环境和特征的调度问题,利用合作博弈理论进行研究,设计求解联盟最优调度的有效算法,并寻求公平稳定的成本分配方法以减少客户成本.

参考文献(References)

- [1] Tang L X, Gong H. A hybrid two-stage transportation and batch scheduling problem[J]. *Applied Mathematical Modelling*, 2008, 32(12): 2467-2479.
- [2] Pei J, Pardalos P M, Liu X B, et al. Serial batching scheduling of deteriorating jobs in a two-stage supply chain to minimize the makespan[J]. *European Journal of Operational Research*, 2015, 244(1): 13-25.
- [3] Behnamian J, Fatemi Ghomi S M T, Jolai F, et al. Realistic two-stage flowshop batch scheduling problems with transportation capacity and times[J]. *Applied Mathematical Modelling*, 2012, 36(2): 723-735.
- [4] Kong L L, Li H, Luo H B, et al. Optimal single-machine batch scheduling for the manufacture, transportation and JIT assembly of precast construction with changeover costs within due dates[J]. *Automation in Construction*, 2017, 81: 34-43.
- [5] 宫华, 张二梅, 刘芳. 传搁时间约束下的运输与批处理机生产协调调度[J]. *控制与决策*, 2017, 32(6): 995-1000.
(Gong H, Zhang E M, Liu F. Coordinated scheduling of transportation and production on batching-machine with tracking time constraint[J]. *Control and Decision*, 2017, 32(6): 995-1000.)
- [6] Liu C H. Using genetic algorithms for the coordinated scheduling problem of a batching machine and two-stage transportation[J]. *Applied Mathematics and Computation*, 2011, 217(24): 10095-10104.
- [7] 舒小娟. 基于强化学习的批处理机生产与运输协调调度研究[D]. 沈阳: 沈阳理工大学, 2020.
(Shu X J. The research on the coordinated scheduling problems of batch machine production and transportation based on reinforcement learning algorithm[D]. Shenyang: Shenyang Ligong University, 2020.)
- [8] 宫华, 许可, 孙文娟. 带尺寸约束的二机流水车间生产运输协调博弈调度问题[J]. *控制与决策*, 2023, 38(7): 1942-1950.
(Gong H, Xu K, Sun W J. Coordinated two-machine flow-shop and transportation scheduling under size constraint and game theory consideration[J]. *Control and Decision*, 2023, 38(7): 1942-1950.)
- [9] 胡乃平, 张立英. 合作博弈在硫化车间生产调度中的应用研究[J]. *计算机与数字工程*, 2016, 44(2): 200-202.
(Hu N P, Zhang L Y. Production scheduling of vulcanization workshop based on cooperative game[J]. *Computer & Digital Engineering*, 2016, 44(2): 200-202.)
- [10] Calleja P, Borm P, Hamers H, et al. On a new class of parallel sequencing situations and related games[J]. *Annals of Operations Research*, 2002, 109(1): 265-277.
- [11] Curiel I, Pederzoli G, Tijs S. Sequencing games[J]. *European Journal of Operational Research*, 1989, 40(3): 344-351.
- [12] Yang G J, Sun H, Uetz M. Cooperative sequencing games with position-dependent learning effect[J]. *Operations Research Letters*, 2020, 48(4): 428-434.
- [13] Hamers H, Borm P, Tijs S. On games corresponding to sequencing situations with ready times[J]. *Mathematical Programming*, 1995, 69(1): 471-483.
- [14] Li F, Yang Y. Cooperation in a single-machine scheduling problem with job deterioration[C]. 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference. Chongqing, 2016: 79-82.
- [15] Borm P, Fiestras-Janeiro G, Hamers H, et al. On the convexity of games corresponding to sequencing situations with due dates[J]. *European Journal of Operational Research*, 2002, 136(3): 616-634.
- [16] Yang G J, Sun H, Hou D S, et al. Games in sequencing situations with externalities[J]. *European Journal of Operational Research*, 2019, 278(2): 699-708.
- [17] Zhou Y P, Gu X S. A new cost allocation approach on one machine sequencing games[J]. *Applied Mechanics and Materials*, 2011, 121/122/123/124/125/126: 3731-3735.
- [18] 孙文娟, 宫华, 许可, 等. 带有交货期的比例流水车间调度问题的合作博弈[J]. *控制与决策*, 2022, 37(3): 712-720.
(Sun W J, Gong H, Xu K, et al. Cooperative games on proportionate flow-shop scheduling problem with due-dates[J]. *Control and Decision*, 2022, 37(3): 712-720.)
- [19] Çiftçi B, Borm P, Hamers H, et al. Batch sequencing and cooperation[J]. *Journal of Scheduling*, 2013, 16(4): 405-415.

作者简介

宫华(1976—),女,教授,博士生导师,从事生产调度与物流优化、深度学习与强化学习等研究, E-mail: gonghua@sylu.edu.cn;

孙红梅(1998—),女,硕士生,从事系统建模与优化控制、优化理论与算法等研究, E-mail: 2056479243@qq.com;

孙文娟(1982—),女,副教授,博士生,从事生产调度与物流优化、优化理论与算法等研究, E-mail: sunwenjuan@sylu.edu.cn;

许可(1982—),女,副教授,博士生,从事优化理论与算法、生产调度与物流优化等研究, E-mail: xuke@sylu.edu.cn.