

控制与决策

Control and Decision

带有交货期的比例流水车间调度问题的合作博弈

孙文娟, 宫华, 许可, 刘鹏

引用本文:

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.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2020.1355>

您可能感兴趣的其他文章

Articles you may be interested in

基于机床超低待机状态的流水车间能耗调度

Energy consumption scheduling in flow shop based on ultra-low idle state of numerical control machine tools

控制与决策. 2021, 36(1): 143–151 <https://doi.org/10.13195/j.kzyjc.2019.0433>

区间数可重入混合流水车间调度与预维护协同优化

Collaborative optimization of interval number reentrant hybrid flow shop scheduling and preventive maintenance

控制与决策. 2021, 36(11): 2599–2608 <https://doi.org/10.13195/j.kzyjc.2020.0973>

基于多班教学优化的多目标分布式混合流水车间调度

Multi-class teaching-learning-based optimization for multi-objective distributed hybrid flow shop scheduling

控制与决策. 2021, 36(2): 303–313 <https://doi.org/10.13195/j.kzyjc.2020.0549>

基于改进蛙跳算法的分布式两阶段混合流水车间调度

An improved shuffled frog leaping algorithm for the distributed two-stage hybrid flow shop scheduling

控制与决策. 2021, 36(1): 241–248 <https://doi.org/10.13195/j.kzyjc.2019.0472>

顺序依赖的调整时间和拖期的无缝钢管热轧批量调度算法

Hot-rolled batch scheduling algorithm for seamless steel tube with sequence-dependent setup times and tardiness

控制与决策. 2021, 36(2): 505–512 <https://doi.org/10.13195/j.kzyjc.2019.0723>

带有交货期的比例流水车间调度问题的合作博弈

孙文娟^{1,2}, 宫 华^{1,2†}, 许 可^{1,2}, 刘 鹏²

(1. 沈阳理工大学 理学院, 沈阳 110159; 2. 沈阳工业大学 管理学院, 沈阳 110870)

摘要: 针对具有多个客户订单的比例流水车间调度问题, 在考虑有交货期及提前和拖期惩罚下, 以客户支出成本为优化指标, 在客户通过合作结成联盟的方式下, 以联盟内成员进行重新调度所获得的最大成本节省为联盟的价值, 建立合作博弈模型. 该合作博弈是具有无外部性的平衡博弈, 从而有非空核. 考虑到客户对提前加工和延迟加工的迫切程度不同, 提出基于提前及拖期惩罚的 β 规则分配方法, 该方法能得到带有交货期的比例流水车间调度合作博弈的一个核分配. 通过混合差分进化算法求解最优调度顺序, 实验结果验证了基于合作博弈模型的调度方法及成本分配方法的有效性.

关键词: 比例流水车间调度; 合作博弈; 混合差分进化算法; 成本分配; 交货期; 核

中图分类号: TP273

文献标志码: A

DOI: 10.13195/j.kzyjc.2020.1355

开放科学(资源服务)标识码(OSID):



引用格式: 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.

Cooperative games on proportionate flow-shop scheduling problem with due-dates

SUN Wen-juan^{1,2}, GONG Hua^{1,2†}, XU Ke^{1,2}, LIU Peng²

(1. School of Science, Shenyang Ligong University, Shenyang 110159, China; 2. School of Management, Shenyang University of Technology, Shenyang 110870, China)

Abstract: For the proportionate flow-shop scheduling problem with multiple customer orders, due dates, earliness and tardiness penalties are taken into consideration. The cooperative games model is established with the customer's expenditure cost as the optimization objective, and the maximum cost savings obtained from the rescheduling as the worth of the coalition, which is formed by customers through cooperation. It is proved that the corresponding cooperative games which have no externality are balanced and have a nonempty core. Taking into account the different urgency degree of customers to advance processing and delay processing, a kind of cost allocation method based on the β rule with earliness and tardiness penalties is proposed, and it is also proved that this allocation method gives a core allocation of the game. Finally, a hybrid differential evolutionary algorithm is designed to solve the optimal scheduling arrangement. The experimental results verify the effectiveness of the scheduling strategy and the cost allocation method based on the cooperative games model.

Keywords: proportionate flow-shop scheduling; cooperative games; hybrid differential evolution algorithm; cost allocation; due-dates; core

0 引言

在工业生产领域中,企业大多采用面向订单的生产模式,必须尽可能考虑自身的生产需求和不同客户不同订单的具体需求,以缩减企业成本,提高企业竞争力.传统的作业车间任务调度以生产企业为主体,在综合考虑加工订单、加工能力及其他约束指标(如交货期、加工成本等)的基础上,进行调度操作达到总

体目标最优^[1-3].传统的调度方法忽略了各加工订单之间的竞争和合作的可能.因此,利用博弈理论来研究生产调度问题已成为国内外学者的研究热点,尤其是利用非合作博弈理论进行生产调度研究^[4-6].

由于加工订单来源于不同的客户,提前完成将产生相应的库存成本,延迟完成又会造成相应的拖期惩罚,客户希望自己的加工订单能够按期完成,而生产

收稿日期: 2020-09-29; 录用日期: 2021-01-08.

基金项目: 辽宁省教育厅科学研究经费项目(LG202025, WJGD2020001); 辽宁省百千万人才工程项目(2019).

责任编委: 王凌.

†通讯作者. E-mail: gonghua@sylyu.edu.cn.

企业生产能力有限,难以满足所有客户需求.因此,客户愿意通过合作的方式结成联盟,在联盟内对订单进行重新排序,以对节省的成本进行合理分配,使得联盟内的客户都能受益,这种合作方式对企业生产运作管理具有重要的实际意义.

在生产调度领域,最早由文献[7]将合作博弈应用到成本分配问题的研究中.文献[8]给出了标准调度模型,证明了待加工工序不存在约束条件,目标函数为加权完工时间的单机调度问题的合作博弈为凸博弈.文献[9-11]分别研究了具有准备时间、交货期以及带有工件恶化的单机调度博弈.文献[12-13]建立了并行机调度的合作博弈模型,证明了在一定条件下,可得出平衡博弈的结果.文献[14]考虑了目标函数为加权完工时间的比例流水车间调度问题,证明了该博弈问题具有非空核.文献[15]针对加工时间和工序相关的一类流水车间调度问题建立了合作博弈模型,证明了此类博弈是平衡博弈,并提出了相应的成本分配方法.文献[16]考虑了多阶段排序博弈,提出了产生核元素的补偿分配规则.文献[17]研究了带有拖期惩罚的单机排序博弈,证明了该博弈具有非空核,并提出了两种核分配方法.文献[18-19]考虑了具有位置相关学习效应及非正外部性的单机排序博弈,给出了相应的核分配方法.

目前,已有的文献主要是利用合作博弈理论研究单机调度,以及成本性能指标为完工时间的线性函数的并行机及流水机调度.本文的工作与以上不同,考虑带有交货期的比例流水车间调度问题(proportionate flow-shop scheduling problem, PFSP).PFSP是一类特殊的流水车间调度问题,其每个任务在不同机器上加工时间相同.本文针对PFSP,考虑更符合实际生产特征的目标函数,将客户支出成本定义为加工成本、提前及拖期惩罚之和,研究客户间以合作的方式结成联盟,建立合作博弈模型优化生产调度,并对节省的成本进行分配.

在利用合作博弈研究生产调度问题时,各联盟的最优调度的求解尤其重要.对于带有交货期的比例流水车间调度问题,目前已有一些学者对其优化算法进行了研究.文献[20]证明了目标为最大化加权准时生产工件数的比例流水车间调度为NP难问题,并提出了近似多项式算法.文献[21]研究了带有公共流及交货期的一类比例流水车间问题,给出了多项式时间算法.文献[22]研究了带有交货期窗口的比例流水车间调度问题,并提出了相应的算法.这些PFSP问题中各工件的拖期惩罚及提前惩罚因子或相

同,或与加工位置有关,而本文考虑的PFSP问题,对于不同任务交货期、成本系数、拖期惩罚及提前惩罚因子均不同且与加工位置无关,求解更为复杂.因此,本文设计了混合差分进化算法(hybrid differential evolution, HDE)来求解此类PFSP的最优调度.

本文的结构如下:第1部分建立了带有交货期的PFSP的数学模型及合作博弈模型,证明了合作博弈具有非空核;第2部分对节省的成本提出了基于提前及拖期惩罚的 β 规则分配方法,证明了该分配方法能得到合作博弈的一个核分配;第3部分设计了求解PFSP的混合差分进化算法;第4部分通过实验仿真对基于合作博弈的调度方法及成本分配方法进行了验证.

1 带有交货期的PFSP的合作博弈模型

1.1 带有交货期的PFSP

本文研究带有交货期的PFSP描述如下:有 n 个待加工任务需要经由 m 道工序加工完成,每道工序由不同的机器加工,各任务的 m 道工序加工路径相同,且同一个任务在所有机器上的加工时间相同.各加工任务在每台机器上的加工时间、成本系数、交货期及未按时交货带来的提前及拖期惩罚已知.并且假设:1)所有待加工任务0时刻可利用;2)任务加工的准备时间包含在加工时间中;3)同一时刻,每一任务只能由一台机器加工,每一机器只能加工一个任务;4)每台机器都不会因故障、维护或其他此类原因而中断.

该问题的调度优化目标是确定在每台机器上各任务的加工顺序,使加工成本及提前或拖期带来的惩罚之和(支出成本)最小.一般而言,对于流水车间调度问题,各任务在各机器上的加工顺序不必相同,若相同,则称此调度为置换调度.本文考虑各加工任务按照置换调度进入每台机器.相关符号及说明见表1.

若订单在交货期完成,则客户支出成本与加工时间成线性关系;若订单提前或拖期,则客户支出成本需要增加额外的提前或拖期惩罚.假设各任务的加工流程为依次通过机器1,机器2, ..., 机器 m ,则客户 $i(i \in N)$ 的支出成本为

$$u(i) = \begin{cases} a_i p_i + e_i(d_i - C_{i,m}), & C_{i,m} < d_i; \\ a_i p_i, & C_{i,m} = d_i; \\ a_i p_i + t_i(C_{i,m} - d_i), & C_{i,m} > d_i. \end{cases} \quad (1)$$

设 π_0 表示初始调度顺序,带有交货期的PFSP可以描述为一个8元组 $(N, M, P, A, D, E, T, \pi_0)$,其数

表1 符号说明

符号	名称及说明
N	加工任务集, $N = \{1, 2, \dots, n\}$
M	加工机器集, $M = \{1, 2, \dots, m\}$
i, k	任务索引, $i, k \in N$
j	机器索引, $j \in M$
p_i	任务 i 在各台机器上的加工时间
P	加工时间集, $P = \{p_i, i \in N\}$
$C_{i,j}$	任务 i 在机器 j 上的完工时间
$u(i)$	客户 i 的支出成本
U	客户的支出成本集, $U = \{u(i), i \in N\}$
a_i	任务 i 的加工成本系数
A	加工成本系数集, $A = \{a_i, i \in N\}$
d_i	任务 i 的交货期
D	任务的交货期集, $D = \{d_i, i \in N\}$
e_i	任务 i 的提前惩罚系数
E	提前惩罚集, $E = \{e_i, i \in N\}$
t_i	任务 i 的拖期惩罚系数
T	拖期惩罚集, $T = \{t_i, i \in N\}$
	调度顺序 $\pi: N \rightarrow N$
π	$\pi(i) = k$ 或 $i = \pi^{-1}(k)$ 表示任务 i 在 π 的第 k 个位置上加工
Π_N	所有任务的所有调度顺序的集合

学模型如下.

1) 目标函数: 最小化支出成本函数

$$\min z = \sum_{i=1}^n u(i). \quad (2)$$

2) 约束条件

$$\sum_{i=1}^n x_{ik} = 1, \quad k = 1, 2, \dots, n; \quad (3)$$

$$\sum_{k=1}^n x_{ik} = 1, \quad i = 1, 2, \dots, n; \quad (4)$$

$$C_{\pi^{-1}(1),1} = \sum_{i=1}^n x_{i1} \cdot p_i; \quad (5)$$

$$C_{\pi^{-1}(1),j} = C_{\pi^{-1}(1),j-1} + p_{\pi^{-1}(1)}, \quad j = 2, 3, \dots, m; \quad (6)$$

$$C_{\pi^{-1}(k),1} = C_{\pi^{-1}(k-1),1} + p_{\pi^{-1}(k)}, \quad k = 2, 3, \dots, n; \quad (7)$$

$$C_{\pi^{-1}(k),j} = \max\{C_{\pi^{-1}(k-1),j}, C_{\pi^{-1}(k),j-1}\} + p_{\pi^{-1}(k)}, \quad k = 2, 3, \dots, n, \quad j = 2, 3, \dots, m; \quad (8)$$

$$\pi(i) = \sum_{k=1}^n k \cdot x_{ik}, \quad i = 1, 2, \dots, n; \quad (9)$$

$$x_{ik} = \begin{cases} 1, & \text{若任务 } i \text{ 在 } \pi \text{ 的第 } k \text{ 位上加工;} \\ 0, & \text{otherwise;} \end{cases} \quad (10)$$

$i, k = 1, 2, \dots, n.$

其中: 式(3)和(4)保证了在调度顺序 π 中各任务出现且仅出现一次; 式(5)为第1台机器上调度顺序 π 中第

1个任务的加工完成时间; 式(6)为机器 j 上调度顺序 π 中第1个任务的加工完成时间, 为其在前一台机器上的加工完成时间与其在机器 j 上的加工时间之和; 式(7)为调度顺序 π 中第 k ($k = 2, 3, \dots, n$) 个任务在机器1上的加工完成时间, 为前一任务的加工完成时间与第 k 个任务在机器1上的加工时间之和; 式(8)为调度顺序 π 中第 k 个任务在机器 j ($j = 2, 3, \dots, n$) 上的加工完成时间; 式(9)描述了调度顺序 π 中第 k 个任务与第 i 个订单待加工任务之间的关系; 式(10)描述了决策变量.

1.2 合作博弈模型的建立

将 n 个订单所属的客户看成 n 个博弈方, 根据客户先到先服务原则, 形成一个初始调度顺序. 客户通过小范围或全体合作结成不同的联盟 S , 并在联盟内通过调整调度顺序, 使得联盟内所有客户的总成本节省最大. 为了公平考虑, 本文假设只有订单顺序为依次相邻的客户才能结成联盟. 当所有客户构成一个大联盟时, 博弈方的策略集为该 n 个订单待加工任务的所有可行调度顺序集 Π_N , 博弈方的收益为客户在各调度顺序下的成本节省. 如果无法形成大联盟, 部分订单相邻客户可能形成小联盟, 则此时需要寻找小联盟内的最优调度顺序.

合作博弈解决的问题是寻找最优调度顺序, 使得联盟内所有客户的总成本节省最大, 并找出合理的节省成本分配方案, 保证联盟的稳定.

定义1 给定一个 n 个博弈方的集合 N , 如果 v 是 N 的所有子集 (有 2^N 个) 上的映射, 即 $v: 2^N \rightarrow \mathbf{R}$ 且 $v(\emptyset) = 0$, 则称有序对 (N, v) 为一个合作博弈, v 为联盟 S ($S \subseteq N$) 对应的特征函数^[23].

定义2 合作博弈 $G = (N, v)$ 中, 如果有 n 维向量 $x = (x_1, x_2, \dots, x_n)$ 满足 $\sum_{i \in N} x_i = v(N)$, 且对所有的联盟 S ($S \subseteq N$) 有 $\sum_{i \in S} x_i \geq v(S)$, 则称 x 为 G 的一个核分配, x 的集合称为合作博弈 G 的核^[23], 记为 $C(v)$, 其中

$$C(v) = \left\{ x \in \mathbf{R}^n \mid \sum_{i \in N} x_i = v(N), \text{ 且 } \sum_{i \in S} x_i \geq v(S), \forall S \subseteq N \right\}. \quad (11)$$

合作博弈的核可能是空集, 也可能是非空集. 若合作博弈的核非空, 则联盟能够稳定存在. 若核非空, 则还需要有对联盟内客户节省的成本进行合理分配的方法, 使得联盟内每个博弈方的收益大于未结成联盟时的收益, 才能保证博弈方不会偏离联盟.

为了定义 PFSP 合作博弈中联盟的价值, 必须确

定 S 的可行调度集. 为此, 先给出连通联盟的定义: 一个联盟 S , 若对于所有的 $i, j \in S$ 及 $k \in N$, 满足 $\pi_0(i) < \pi_0(k) < \pi_0(j)$ 时, $k \in S$ 成立, 则称联盟 S 关于调度顺序 π_0 是连通的. 若连通联盟 $T \subset S$, 满足对每一个 $i \in S \setminus T, T \cup \{i\}$ 关于调度顺序 π_0 均不连通, 则称 T 为联盟 S 关于调度顺序 π_0 的一个 π_0 -组, 联盟 S 的 π_0 -组集合记为 S/π_0 . 给定一个联盟 S , 一个调度顺序 π , 如果满足: 1) $S/\pi_0 = S/\pi$; 2) $C_{i,m}^\pi = C_{i,m}^{\pi_0}, \forall i \in N \setminus S$. 则称 π 为联盟 S 关于调度顺序 π_0 的一个可行调度顺序. 联盟 S 的所有可行调度顺序集记作 Π_S .

设 π_0 表示初始调度顺序, 带有交货期的 PFSP 对应的合作博弈模型记为 (N, v) , 其中

$$v(S) = \max_{\pi \in \Pi_S} \left(\sum_{i \in S} u_{\pi_0}(i) - \sum_{i \in S} u_\pi(i) \right), \quad (12)$$

其中: $v(S)$ 表示联盟 S 的价值, 为 S 内成员通过合作得到的最大总成本节省值. 因此, 本文研究的带有交货期的 PFSP 可转化为最大化合作博弈联盟总成本节省的问题.

流水车间调度问题中的外部性是指联盟 S 内客户调整调度顺序会影响 S 后客户的任务完工时间, 从而会影响其成本. 下面的引理将证明本文所考虑的比例流水车间合作博弈不具有外部性. 为了叙述方便, 引入如下记号.

给定调度顺序 π , 定义 $F(\pi, i)$ 为排在 i 前面的客户集合, 即 $F(\pi, i) = \{j \in N | \pi(j) < \pi(i)\}$; $\bar{F}(\pi, i)$ 为排在 i 前面的客户加上 i 的集合, 即 $\bar{F}(\pi, i) = F(\pi, i) \cup \{i\}$; $B(\pi, i)$ 为排在 i 后面的客户集合, 即 $B(\pi, i) = \{j \in N | \pi(j) > \pi(i)\}$; $\bar{B}(\pi, i)$ 为排在 i 后面的客户加上 i 的集合, 即 $\bar{B}(\pi, i) = B(\pi, i) \cup \{i\}$.

引理 1 带有交货期的 PFSP 合作博弈 (N, v) 无外部性.

证明 要证明合作博弈 (N, v) 无外部性, 即要证明任意联盟的形成不会影响联盟外的成员. 显然, 任意联盟 $S (S \subseteq N)$ 内客户的合作不会影响 S 前客户加工任务的完工时间, 从而不会影响其支出成本. 而在本文所考虑的 PFSP 中, 任务是按照置换调度顺序进行加工, 故在调度顺序 π 下, 对于每个加工任务 $i \in N$, 其在第 m 台机器上的完工时间为

$$C_{i,m}^\pi = \sum_{j \in \bar{F}(\pi, i)} p_j + (m-1) \max_{j \in \bar{F}(\pi, i)} \{p_j\}. \quad (13)$$

交换联盟 S 内任意任务的加工顺序, S 内最后一个任务的完工时间不变, 从而不会影响 S 后任务的完工时间, 亦不会影响其支出成本. \square

由引理 1 可知, 对于联盟 S 而言, 在联盟 S 的 π_0 -

组内进行重排序, 其他任务加工位置不变, 所得到的调度顺序都是 S 关于 π_0 的可行调度顺序.

定义 3 设合作博弈 $G = (N, v)$, 若对任意联盟 $S, T \subset N$, 且 $S \cap T = \emptyset$, 有 $v(S) + v(T) \leq v(S \cup T)$, 则称 G 是超可加博弈^[23].

定理 1 PFSP 合作博弈是超可加博弈.

证明 设联盟 S 和 T 的最优调度顺序分别为 π_S 和 π_T , 则联盟 S 和 T 内的客户合作带来的成本节省之和满足

$$\begin{aligned} v(S) + v(T) &= \sum_{i \in S} (u_{\pi_0}(i) - u_{\pi_S}(i)) + \sum_{i \in T} (u_{\pi_0}(i) - u_{\pi_T}(i)) = \\ &= \sum_{i \in S \cup T} (u_{\pi_0}(i) - u_{\pi'_{(S \cup T)}}(i)) \leq \\ &= \max_{\pi \in \Pi_{(S \cup T)}} \sum_{i \in S \cup T} (u_{\pi_0}(i) - u_\pi(i)) = \\ &= v(S \cup T). \end{aligned} \quad (14)$$

其中: $\pi'_{(S \cup T)}$ 表示联盟 S 及 T 中的调度顺序分别与 π_S 及 π_T 一致, 其他任务调度顺序同 π_0 . $\Pi_{(S \cup T)}$ 表示联盟 S 和 T 的并集构成的联盟的所有可行调度顺序, 显然 $\pi'_{(S \cup T)} \in \Pi_{(S \cup T)}$, 式(14)中不等式成立. 由定义 3 可知, PFSP 合作博弈是超可加博弈. \square

定理 2 PFSP 合作博弈是平衡博弈, 具有非空核.

证明 由引理 1 及定理 1 易知, PFSP 合作博弈满足以下 3 个条件:

- 1) 对于每个客户 $i \in N, v(\{i\}) = 0$;
- 2) 合作博弈 (N, v) 是超可加的;
- 3) 对于所有的 $S (S \subseteq N), v(S) = \sum_{T \in S/\pi_0} v(T)$.

满足以上 3 个条件的合作博弈是 π_0 -组可加博弈, Le Breton 等^[24]证明了此类博弈是平衡博弈, 且核非空. \square

2 基于提前及拖期惩罚的 β 规则分配方法

本文研究的带有交货期的 PFSP 合作博弈, 由于形成联盟可节省成本, 当初始调度顺序不是大联盟下的最优调度顺序时, 可能形成联盟. 但联盟能否真正形成, 还需寻找能够使所有博弈方满意的成本分配方案, 使得联盟内的每个客户都不会偏离.

Curjel 等^[23]在研究排列博弈时提出了 β 规则分配方法, 在可行调度范围内通过对博弈方 i 加入前面客户集及加入后面客户集产生的边际成本进行加权求和. 本文在 β 规则分配方法的基础上, 考虑到客户提前及拖期惩罚因子不同, 对提前加工或延后加工的迫切程度也不同, 提出了基于提前及拖期惩罚的 β 规

则分配方法.

在初始排列 π_0 中, i 加入 i 前面客户集合产生的边际成本为 $v(\bar{F}(\pi_0, i)) - v(F(\pi_0, i))$, i 加入 i 后面客户集合产生的边际成本为 $v(\bar{B}(\pi_0, i)) - v(B(\pi_0, i))$, 则客户 i 获得的成本分配为

$$x_i(v) = \beta_{et}[v(\bar{F}(\pi_0, i)) - v(F(\pi_0, i))] + (1 - \beta_{et})[v(\bar{B}(\pi_0, i)) - v(B(\pi_0, i))]. \quad (15)$$

式(15)即为基于提前和拖期惩罚的 β 规则分配方法, 其中

$$\beta_{et} = \frac{\sum_{i=1}^n e_i}{\sum_{i=1}^n (e_i + t_i)}. \quad (16)$$

定理3 基于提前和拖期惩罚的 β 规则分配方法得到的成本节省分配是带有交货期的 PFSP 合作博弈的一个核分配.

证明 设初始排列为 $\pi_0 = (1, 2, \dots, n)$. 根据定义有 $\bar{F}(\pi_0, n) = \bar{B}(\pi_0, 1) = N$, $F(\pi_0, 1) = B(\pi_0, n) = \emptyset$, $F(\pi_0, i) = \bar{F}(\pi_0, i - 1)$, $B(\pi_0, i - 1) = \bar{B}(\pi_0, i)$. 因此, 在大联盟中, 所有客户获得成本分配之和为

$$\begin{aligned} \sum_{i \in N} x_i(v) &= \beta_{et}[v(\bar{F}(\pi_0, 1)) - v(F(\pi_0, 1))] + \\ &v(\bar{F}(\pi_0, 2)) - v(F(\pi_0, 2)) + \dots + \\ &v(\bar{F}(\pi_0, n)) - v(F(\pi_0, n))] + \\ &(1 - \beta_{et})[v(\bar{B}(\pi_0, 1)) - v(B(\pi_0, 1))] + \\ &v(\bar{B}(\pi_0, 2)) - v(B(\pi_0, 2)) + \dots + \\ &v(\bar{B}(\pi_0, n)) - v(B(\pi_0, n))] = \\ &\beta_{et}v(\bar{F}(\pi_0, n)) + (1 - \beta_{et})v(\bar{B}(\pi_0, 1)) = \\ &\beta_{et}v(N) + (1 - \beta_{et})v(N) = v(N). \end{aligned}$$

即所有成本节省被全部分配给 N 中各客户.

又因为, 对于任意 $S \subset N$, $v(S) = \sum_{T \in S/\pi_0} v(T)$.

设 $T = \{k, T_1, t\}$, 其中 k, t 分别为 T 中序号最小和最大的客户, $T_1 \subset T$, 则由于合作博弈 (N, v) 是超可加的, 从而有

$$\begin{aligned} \sum_{i \in S} x_i &= \sum_{T \in S/\pi_0} \sum_{i \in T} \{\beta_{et}[v(\bar{F}(\pi_0, i)) - v(F(\pi_0, i))] + \\ &(1 - \beta_{et})[v(\bar{B}(\pi_0, i)) - v(B(\pi_0, i))]\} = \\ &\sum_{T \in S/\pi_0} \{\beta_{et}[v(\bar{F}(\pi_0, t)) - v(F(\pi_0, k))] + \end{aligned}$$

$$\begin{aligned} &(1 - \beta_{et})[v(\bar{B}(\pi_0, k)) - v(B(\pi_0, t))]\} \geq \\ &\sum_{T \in S/\pi_0} \{\beta_{et}v(T) + (1 - \beta_{et})v(T)\} = \\ &\sum_{T \in S/\pi_0} v(T) = v(S). \end{aligned}$$

综上可知, 基于提前和拖期惩罚的 β 规则分配方法所得到的分配 x 满足式(11), 该分配是带有交货期的 PFSP 合作博弈的一个核分配. \square

3 基于HDE的合作博弈模型的求解

基于提前和拖期惩罚的 β 规则分配方法能够得到带有交货期的 PFSP 合作博弈的一个核分配. 定理3表明, 按照该方法对联盟内客户的成本节省进行分配是公平合理的, 此时联盟能够稳定存在. 如何寻找各联盟内的最优调度, 使得联盟成本节省最大, 是合作博弈需要解决的关键. 本文考虑到 PFSP 合作博弈中成本函数的复杂性, 设计混合差分进化算法(HDE)来求解各联盟内的最优调度.

差分进化算法(DE)是一种人工智能优化算法, 在每一个新个体的生成过程中, DE利用了父代多个个体间的差分向量实现变异, 其变异概率高于遗传算法, 因而, 搜索能力更强, 速度更快. 然而, DE的进化机制主要依赖于生成多个新个体后的选择过程, 对于参与繁殖的父代个体又进行预先的优化处理, 所以计算效率较低. 遗传算法(GA)以其全局寻优、鲁棒性强等特点, 被广泛应用于各个领域, 但在进化后, 收敛速度较慢, 容易陷入局部最优. 结合所建立的合作博弈模型特点, 本文设计了将DE与GA相结合的混合差分进化算法求解不同联盟下的最优调度. 利用DE框架结合GA变异及交叉机制, 既保持了遗传算法强大的全局搜索能力, 又加快了搜索速度, 从而提升算法的性能.

3.1 编码

针对本文所研究的 PFSP, HDE采用顺序编码表示待优化问题的解, 种群个体由任务向量组成. 若 n 表示需要加工的任务总数, 则该问题的解可以用 $1 \sim n$ 的排序来表示. 如个体编码(6, 3, 7, 5, 1, 2, 4, 8, 9)表示任务排序为6、3、7、5、1、2、4、8、9, 其中个体编码第1位的位值为6, 最后一位位值为9, 表示首先加工任务6, 最后加工任务9.

3.2 变异

本文采用3种变异算子: 差分变异、翻转变异和插入变异. 随机选择一种算子进行变异操作, 这样既可以充分利用差分进化算法向整个群体学习的能力, 又增加了种群多样性.

3.2.1 差分变异

采用DE/rand/1/bin变异策略

$$v_i^{t+1} = x_{r_1}^t + F(x_{r_2}^t - x_{r_3}^t). \quad (17)$$

其中: $x_{r_1}^t, x_{r_2}^t, x_{r_3}^t$ 分别为在第 t 代种群中随机选取的3个不同的父代个体, F 为缩放因子, v_i 为变异后产生的新个体.

需要注意的是,在进行差分变异后,变异个体的编码可能不再是任务向量.如经过差分变异操作后,变异个体编码为(2.83, 1.57, 3.6, 5.5, 4.2, 1.6, 5.6, 6.2, 6.83),此时需要将其转化为顺序编码.本文采用随机转换规则,即按照每个个体位值的大小进行顺序编码.此变异个体编码中,位值最小为1.57,对应的加工任务为1;最大为6.83,对应的加工任务为9.以此类推,则按照位值大小排序后,顺序编码为:(3, 1, 4, 6, 5, 2, 7, 8, 9).

3.2.2 翻转变异

在翻转变异中进行翻转的数值长度和位置均随机选取,具体操作为:在编码序列中随机选取两个点,对其间的数值进行逆转.如在个体(6, 3, 7, 5, 1, 2, 4, 8, 9)中,随机选取“7”及“2”,逆转后变为“2, 1, 5, 7”,(6, 3, 2, 1, 5, 7, 4, 8, 9).

3.2.3 插入变异

在任务排序中随机选取一个编码数值,将其插入到另一个编码数值之前.如在个体(6, 3, 7, 5, 1, 2, 4, 8, 9)中随机选取任务3,将其插入到任务8与任务9之间,并将任务6与任务7连接在一起,形成一个新的个体(6, 7, 5, 1, 2, 4, 3, 9).

3.3 交叉

采用Goldberg等^[25]提出的PMX(部分映射交叉)方法进行交叉操作,PMX交叉流程如图1所示.

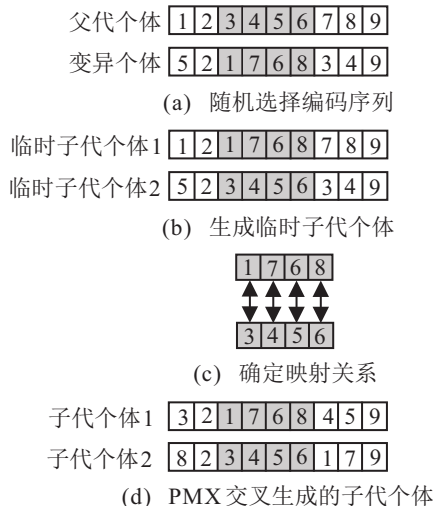


图1 PMX交叉

在交叉操作之前,需要进行判定,若满足随机数 $\text{rand}(0, 1)$ 小于交叉概率 CR , 则进行PMX交叉, 否则不进行,其具体步骤如下.

step 1: 随机选择一个父代个体和变异个体中一段连续的编码序列,其中编码序列的长度随机,且被选位置相同,如图1(a)灰色部分.

step 2: 交换这两个个体所选位置上的编码,生成两个临时子代个体,如图1(b).

step 3: 冲突检测,依据所选的编码序列的编码数值建立一个映射关系.在图1(c)中,以映射关系 $8 \leftrightarrow 6 \leftrightarrow 5$ 为例,可以看到step 2中临时子代个体1存在两个8,这时将子串外的编码8通过映射关系转变为5.依次类推,直到子代个体中没有冲突为止.

step 4: 最终生成子代个体,如图1(d).

3.4 选择

本文采用贪心选择的思路将得到的子代个体与父代目标个体进行比较,将适应度值较大的个体保存到下一次迭代过程中.由于所研究的目标是寻找最优调度顺序使所有客户总支出成本节省最大,此处适应度值应是对应的调度顺序下的所有客户总成本节省.

HDE算法具体流程如下.

step 1: 种群初始化,设置HDE算法的参数.种群数目 N_p ,交叉概率 CR ,变异缩放因子 F ,最大迭代次数 MI .初始迭代次数 $t = 0$,随机产生的初始种群为 $x^0 = \{x_1^0, x_2^0, \dots, x_{N_p}^0\}$.

step 2: 定义当前目标个体索引号 $i = 1$.

step 3: 在目标个体 x_i^t 之外随机选择个体 $x_{r_1}^t, x_{r_2}^t, x_{r_3}^t$,进行变异操作,从3种变异算子中,随机选择一种变异算子生成一个变异个体.

step 4: 若满足 $\text{rand}(0, 1) < \text{CR}$, 则进行交叉操作,对 x_i^t 和 v_i^{t+1} 执行PMX交叉生成新个体 u_i^{t+1} ; 否则,不进行交叉操作,将 x_i^t 赋值给 u_i^{t+1} .

step 5: 执行选择操作,若新个体 u_i^{t+1} 优于目标个体 x_i^t , 则 $x_i^{t+1} = u_i^{t+1}$; 否则, $x_i^{t+1} = x_i^t$.

step 6: 判断是否满足 $i \geq N_p$, 若满足,则进行step 7; 若不满足,则更新 $i = i + 1$, 返回step 3, 继续按照上述操作执行.

step 7: 判断是否满足算法终止条件,即 $t \geq \text{MI}$, 若满足,则迭代结束输出最优解; 否则, $t = t + 1$, 返回step 2, 继续按照上述操作执行.

算法具体流程如图2所示.

运用HDE求合作博弈最优调度时,联盟外客户的任务按初始调度顺序编号后,不再参与变异、交叉

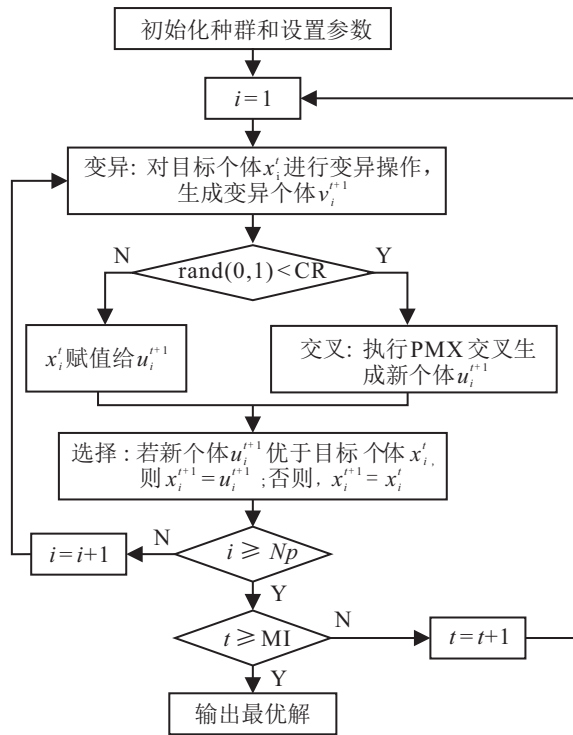


图2 HDE算法流程

操作,即其在调度顺序上的位置始终保持不变,保证了求得的最优调度为可行调度,遵循了合作博弈的公平原则.此外,当联盟内成员个数 $k = 2$ 时,由于差分

变异必须要求具有3个以上不同的个体,可直接按照全排列生成2个个体,通过贪婪选择选出最优调度顺序.

3.5 HDE的性能分析

为验证HDE求解最优调度的性能,首先将HDE通过求解OR-Library中目标函数为 C_{max} 的Car类例题进行测试,将实验结果与标准差分进化算法(DE)、文献[26]和文献[27]中的强化学习(QL)及萤火虫算法(FA)的求解结果进行比较.算法程序采用Matlab 2017a编写,运行环境为Windows7 64位系统,处理器为1.9 GHz,16 GB内存.算法参数设置为:HDE中,最大迭代次数 $MI = 300$,种群数目 $Np = 40$,交叉概率 $CR = 0.2$,差分变异缩放因子 $F = 0.5$.在FA算法中, $MI = 300$,种群数目为40.在QL中,迭代次数为5000.每种算法对Car类问题独立运行20次,测试结果如表2所示.其中: BRE为最优相对误差 ($BRE = (\min(C_{max} - C^*)/C^* \times 100\%)$), ARE为平均相对误差 ($ARE = (\text{avg}(C_{max} - C^*)/C^* \times 100\%)$), WRE为最差相对误差 ($WRE = (\max(C_{max} - C^*)/C^* \times 100\%)$), C_{max} 为算法所求的最小最大完工时间, C^* 为对应调度问题实际最小最大完工时间.

表2 Car类问题测试结果

实验组	C^*	n, m	BRE				ARE				WRE			
			HDE	DE	FA	QL	HDE	DE	FA	QL	HDE	DE	FA	QL
Car 1	7038	11,5	0	0	0	0	0	0.32	0	0	0	2.8	0	-
Car 2	7166	13,4	0	1.22	0	0	0	4.57	1.29	0.71	0	6.62	2.93	-
Car 3	7312	12,5	0	1.88	0	1.19	0.75	3.69	1.86	1.91	1.2	5.64	3.16	-
Car 4	8003	14,4	0	0	0	0	0	2.28	0.33	1.12	0	5.24	1.57	-
Car 5	7720	10,6	0	0	0	0	0.06	0.27	0.59	0.61	0.23	0.75	1.45	-
Car 6	8505	8,9	0	0	0	0	0	1.06	0.57	0.94	0	2.47	2.15	-
Car 7	6590	7,7	0	0	0	0	0	0.05	0.04	0	0	0.65	0.80	-
Car 8	8366	8,8	0	0	0	0	0.04	0.41	0.28	0.31	0.15	1.06	1.35	-

由表2可以看出,关于所选的Car类问题,本文所提出的算法均能找到最优解,寻优成功率为100%,高于对应迭代相同次数的DE及FA.同时各项指标也高于新型调度算法QL,说明了HDE具有较好的全局收敛能力,是解决经典置换流水车间调度问题的一种有效算法.由于所研究的带有交货期的PFSP问题合作博弈模型中的性能指标为联盟的总成本节省值,它是各任务完工时间的非线性函数,且对于不同客户,拖期及提前惩罚因子均不同.当问题规模较大时,求解比较困难,目前对这类问题博弈模型的求解算法也较少.因此,本文利用HDE来求解PFSP合作博弈中各

联盟的最优调度.

4 实验仿真

本节采用如下实例对所研究的基于PFSP合作博弈模型的调度方法及基于提前和拖期惩罚的 β 规则分配方法进行验证.

4.1 初始化条件和参数

设有9个订单隶属于9个不同的客户,每个订单有1个待加工任务,每个任务有4道工序需要在4台不同的机器上以相同的顺序加工.相关的初始化参数如表3所示.

表3 相关参数表

任务	1	2	3	4	5	6	7	8	9
<i>P</i>	52	35	50	51	46	50	56	47	50
<i>A</i>	9	8	7	6	5	6	7	8	9
<i>D</i>	150	170	250	300	260	280	150	300	350
<i>E</i>	1.2	0.3	1.4	0.5	1.3	0.6	1.7	1.3	0.3
<i>T</i>	0.4	1.8	0.6	0.5	0.4	1.4	0.8	0.7	0.8

根据客户提交订单的顺序及先到先服务原则, 确定初始调度顺序 $\pi_0 = (1, 2, 3, 4, 5, 6, 7, 8, 9)$.

4.2 实验结果与分析

在初始调度顺序下, 所有客户的总支出成本为 $\sum_{i=1}^9 u_{\pi_0}(i) = 3544.7$.

若部分订单顺序相邻的客户合作结成联盟, 并在联盟内相互转让优先加工权, 则可能获得支出成本的节省. 利用HDE计算各联盟 *S* 的成本节省 $v(S)$, 结果如表4所示.

表4 各联盟的成本节省

联盟 <i>S</i>	成本节省 $v(S)$	联盟 <i>S</i>	成本节省 $v(S)$	联盟 <i>S</i>	成本节省 $v(S)$
12	21.9	567	69.3	45678	124.3
23	15.1	678	12.2	56789	77.1
34	0	789	0	123456	84.7
45	5.9	1234	21.9	234567	382.9
56	8	2345	62.7	345678	391.2
67	12.2	3456	47.3	456789	124.3
78	0	4567	107.6	1234567	382.9
89	0	5678	77.1	2345678	466.5
123	21.9	6789	12.2	3456789	391.2
234	15.1	12345	62.7	12345678	466.5
345	27.1	23456	84.7	23456789	471
456	19.7	34567	337.1	123456789	471

由表4可知, HDE算法求出的各联盟最大成本节省符合合作博弈的超可加性质. 最大成本节省为471, 此时客户2、3、4、5、6、7、8、9合作结成联盟. 最优调度对应的甘特图如图3所示, 最优调度为 $\pi = (1, 7, 5, 6, 8, 4, 3, 9, 2)$, 总支出成本为3073.7.

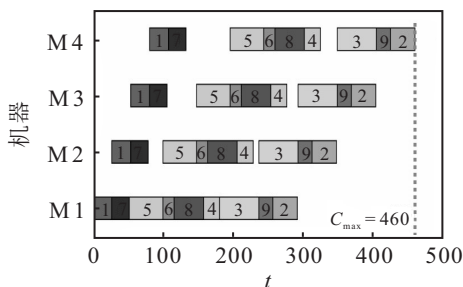


图3 最优调度甘特图

根据式(15)和(16), 运用本文提出的基于提前和

拖期惩罚的 β 规则分配方法, 对节省的成本进行分配, 结果为

$$x(v) = (0, 48.6788, 123.4413, 21.8300, 51.9462, 17.4675, 160.2852, 44.9350, 2.4187).$$

显然 $\sum_{i \in N} x_i(v) = v(N) = 471$, 说明客户合作所获得的成本节省按照基于提前和拖期惩罚的 β 规则分配方法能够全部分配给所有参与的客户, 没有剩余. 在最优调度中, 客户1的位置和初始相同, 对联盟没有贡献, 因此节省成本分配为0; 客户7分到的成本最大, 因为凡是7参加的联盟, 都能获得较大的成本节省. 在此分配方法下, 任意联盟 *S* 中的客户分配得到的成本之和均大于等于联盟的成本节省 $v(S)$, 进一步验证了该分配是一个核分配.

5 结论

本文基于合作博弈理论, 针对具有多个客户订单且带有交货期的比例流水车间调度问题进行了研究. 以加工成本、提前惩罚及拖期惩罚之和构成的客户支出成本为优化指标, 以联盟成员通过合作进行重调度所能达到的最大成本节省值作为联盟对应的特征函数, 建立了带有交货期的PFSP合作博弈模型, 证明了该合作博弈无外部性且具有非空核. 针对不同客户对提前加工或延后加工的迫切程度不同, 提出了基于提前及拖期惩罚的 β 规则分配方法, 并证明在此方法下的成本分配是合作博弈的一个核分配, 能够保证联盟的稳定性. 针对带有交货期的PFSP, 利用DE框架结合GA变异及交叉机制, 设计了混合差分进化算法, 用以求解各联盟内的最优调度. 通过实验验证了混合差分进化算法及成本分配方法的有效性. 未来研究工作可考虑客户带有多加工任务、多机器的复杂车间生产环境调度的合作博弈问题, 并设计多智能体博弈等有效算法求解博弈问题.

参考文献(References)

- [1] Wang L, Wang S Y, Liu M. A pareto-based estimation of distribution algorithm for the multi-objective flexible job-shop scheduling problem[J]. International Journal of Production Research, 2013, 51(12): 3574-3592.
- [2] Mohan J, Lanka K, Rao A N. A review of dynamic job shop scheduling techniques[J]. Procedia Manufacturing, 2019, 30: 34-39.
- [3] Vela C R, Afsar S, Palacios J J, et al. Evolutionary tabu search for flexible due-date satisfaction in fuzzy job shop scheduling[J]. Computers & Operations Research, 2020, 119: 104931.
- [4] Sun D H, He W, Zheng L J et al. Scheduling flexible job shop problem subject to machine breakdown with game

- theory[J]. *International Journal of Production Research*, 2014, 52(13): 3858-3876.
- [5] Han Z H, Zhu Y H, Ma X F, et al. Multiple rules with game theoretic analysis for flexible flow shop scheduling problem with component altering times[J]. *International Journal of Modelling, Identification and Control*, 2016, 26(1): 1-17.
- [6] Zhang Y F, Wang J, Liu Y. Game theory based real-time multi-objective flexible job shop scheduling considering environmental impact[J]. *Journal of Cleaner Production*, 2017, 167: 665-679.
- [7] Tijs S H, Driessen T S H. Game theory and cost allocation problems[J]. *Management Science*, 1986, 32(8): 1015-1028.
- [8] Curiel I, Pederzoli G, Tijs S. Sequencing games[J]. *European Journal of Operational Research*, 1989, 40(3): 344-351.
- [9] Hamers H, Borm P, Tijs S. On games corresponding to sequencing situations with ready times[J]. *Mathematical Programming*, 1995, 69(1/2/3): 471-483.
- [10] 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.
- [11] Li F, Yang Y. Cooperation in a single-machine scheduling problem with job deterioration[C]. *The 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*. Chongqing, 2016: 79-82.
- [12] Calleja P, Borm P, Hamers H, et al. On a new class of parallel sequencing situations[J]. *Annals of Operations Research*, 2002, 109(1/2/3/4): 265-277.
- [13] Slikker M. Balancedness of sequencing games with multiple parallel machines[J]. *Annals of Operations Research*, 2005, 137(1): 177-189.
- [14] Arantza E F, Manuel A M, Borm P, et al. Proportionate flow shop games[J]. *Journal of Scheduling*, 2008, 11(6): 433-447.
- [15] 周艳平, 顾幸生. 一类流水车间调度问题的合作博弈[J]. *化工学报*, 2010, 61(8): 1983-1987.
(Zhou Y P, Gu X S. Cooperation games on a type of flow shop scheduling problem[J]. *CIESC Journal*, 2010, 61(8): 1983-1987.)
- [16] Curiel I. Compensation rules for multi-stage sequencing games[J]. *Annals of Operations Research*, 2015, 225(1): 65-82.
- [17] Zhou Y P, Gu X S. One machine sequencing game with lateness penalties[J]. *International Journal on Information*, 2012, 15(11): 4429-4434.
- [18] Yang G J, Sun H, Uetz M. Cooperative sequencing games with position-dependent learning effect[J]. *Operations Research Letters*, 2020, 48(4): 428-434.
- [19] 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.
- [20] Shabtay D. The just-in-time scheduling problem in a flow-shop scheduling system[J]. *European Journal of Operational Research*, 2012, 216(3): 521-532.
- [21] Mor B, Mosheiov G. Minsum and minmax scheduling on a proportionate flowshop with common flow-allowance[J]. *European Journal of Operational Research*, 2016, 254(2): 360-370.
- [22] Sun X Y, Geng X N, Liu T. Due-window assignment scheduling in the proportionate flow shop setting[J]. *Annals of Operations Research*, 2020, 292(1): 113-131.
- [23] Curiel I, Potters J, Prasad R, et al. Sequencing and cooperation[J]. *Operations Research*, 1994, 42(3): 566-568.
- [24] Le Breton M, Owen G, Weber S. Strongly balanced cooperative games[J]. *International Journal of Game Theory*, 1992, 20(4): 419-427.
- [25] Goldberg D, Lingle R. Allelesloci and the traveling salesman problem[C]. *Proceedings of 1st International Conference on Genetic Algorithms*. Hillsdale, 1985: 154-159.
- [26] 张东阳, 叶春明. 应用强化学习算法求解置换流水车间调度问题[J]. *计算机系统应用*, 2019, 28(12): 195-199.
(Zhang D Y, Ye C M. Reinforcement learning algorithm for permutation flow shop scheduling to minimize makespan[J]. *Computer Systems & Applications*, 2019, 28(12): 195-199.)
- [27] 刘长平, 叶春明. 置换流水车间调度问题的萤火虫算法求解[J]. *工业工程与管理*, 2012, 17(3): 56-59.
(Liu C P, Ye C M. Solving permutation flow shop scheduling problem by firefly algorithm[J]. *Industrial Engineering and Management*, 2012, 17(3): 56-59.)

作者简介

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

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;

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的研究, E-mail: xuke@sylu.edu.cn;

刘鹏(1978-),男,教授,博士生导师,从事生产调度与物流优化等研究, E-mail: liup7802@163.com.

(责任编辑: 孙艺红)