

DOI: 10.3901/JME.2022.09.210

# 含私有信息的多代理作业车间协商调度算法<sup>\*</sup>

孙树栋<sup>1,2</sup> 周新民<sup>1,2</sup> 常昇博<sup>1,2</sup>

(1. 西北工业大学机电学院 西安 710072;

2. 西北工业大学工业工程与智能制造工业和信息化部重点实验室 西安 710072)

**摘要:** 在考虑用户互为竞争对手、不愿共享其目标的情况下, 用户与车间共同组成调度决策群体, 形成代理目标为私有信息的多代理作业车间调度问题。以最大化群体社会福利为优化目标, 构建了多代理作业车间调度问题的数学模型。设计了遗传进化-评分决策两阶段协商调度机制, 并提出了相应的两阶段协商调度算法。在不披露代理目标的前提下, 遗传进化阶段采用帕累托优化排序算法, 生成非支配调度方案集; 评分决策阶段采用基于效用的线性转换评分算法, 实现从非支配方案集中选出社会福利较优的调度方案。大量仿真研究表明, 所提遗传进化-评分决策两阶段协商调度算法, 其整体性能优于现有协商调度算法, 能够产生社会福利更高的调度方案。

**关键词:** 多代理调度; 作业车间; 协商机制; 帕累托优化; 遗传算法

中图分类号: TP278

## Negotiation Scheduling Algorithm for Multi-agent Job Shop with Private Information

SUN Shudong<sup>1,2</sup> ZHOU Xinmin<sup>1,2</sup> CHANG Shengbo<sup>1,2</sup>

(1. School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an 710072;

2. Key Lab of Industrial Engineering and Intelligent Manufacturing, Northwestern Polytechnical University, Xi'an 710072)

**Abstract:** This research focuses on the multi-agent job shop scheduling problem with private information, where the users are competitors and unwilling to share their objectives, and the users and the job shop make up a decision group. Firstly, a multi-agent job shop scheduling model intending to maximize social welfare is set up. Then, a two-stage negotiation scheduling algorithm is proposed which is composed of genetic evolution and scoring decision stages. In order to avoid revealing private information, a ranking vote-based Pareto sorting algorithm is proposed for the genetic evolution stage, and a score algorithm with linear conversion based on utility is used in the decision stage. Finally, a large number of multi-agent job shop scheduling instances are constructed based on the job shop scheduling benchmark problems to analyze the performance of the proposed algorithm. The simulation results have shown that the overall performance of the proposed two-stage negotiation scheduling algorithm is better than the existing scheduling algorithms, and can get a schedule with higher social welfare.

**Key words:** multi-agent scheduling; job shop; negotiation mechanism; Pareto optimization; genetic algorithm

## 0 前言

随着技术的进步和竞争的加剧, 传统的车间生产模式已经从规模化生产快速地转向个性化定制的服务模式, 这种模式导致用户驱动的个性化生产应运而生。在个性化生产环境下, 作业车间调度的优

化目标将由原来主要考虑车间资源优化目标转向同时考虑各个用户的差异化需求、满足每个用户的个性化目标; 且由于用户之间的竞争特性, 用户不会轻易披露自身目标信息。若将各个用户和车间看成不同代理, 便产生了一种拥有私有信息的多代理作业车间调度新问题。

已有研究大多采用集中式优化来求解多代理调度问题, 集中式优化可以获得质量较高的调度方案, 然而要求所有代理的调度目标必须完全公开<sup>[1-3]</sup>。针

<sup>\*</sup> 国家自然科学基金资助项目(51975482)。20210524 收到初稿, 20210905 收到修改稿

对此类问题, 采用分布式优化, 特别是协商调度, 是一个更为合理的选择<sup>[1-3]</sup>。它允许拥有私有信息的多个代理共同参与调度决策, 并通过代理间的竞争和协调, 产生满足各代理需求并具有较高社会福利的作业车间调度方案<sup>[1-6]</sup>。

根据进化与决策逻辑的不同, 现有协商调度算法主要分为进化决策交替协商调度<sup>[1, 3-5]</sup>和进化-决策两阶段协商调度<sup>[2]</sup>。进化决策交替协商调度在每个协商轮次, 代理都通过投票的方式选择一个拟定方案, 然后各代理以此方案为基础生成新方案, 重复此过程便可以不断使该拟定方案进化。为保证收敛性, 现有研究通常基于元启发式算法设计, 主要有基于模拟退火的协商调度算法<sup>[1, 5]</sup>、基于遗传算法的协商调度算法<sup>[3]</sup>等。该调度策略采用单点局部优化方法, 通过对单个拟定方案进行多次的局部优化, 使其接近帕累托前沿。但是, 由于单点局部优化无法广泛的搜索帕累托前沿, 导致拟定方案有向帕累托前沿中社会福利较差的区域进化的可能。此外, 在协商过程中的拟定方案接近帕累托前沿时, 由于代理间存在竞争, 协商会陷入停滞, 无法取得较好的社会福利。

进化-决策两阶段协商调度在进化阶段的每次协商中, 都生成一定数目的新方案集, 代理通过排序投票更新非支配方案集, 重复此过程便可不断使非支配方案集进化。进化完成后, 通过投票等方式协调代理偏好, 并从所得非支配方案集中选择一个方案作为协商结果。与进化决策交替协商调度不同, 其进化阶段采用多点全局优化的方法, 通过对多点进行迭代优化, 能够得到接近帕累托前沿的非支配方案集, 决策阶段比较各非支配方案的社会福利, 从中选出一个社会福利较大的调度方案。

## 1 调度模型

设共有  $n$  个工件分属于  $u \geq 2$  个用户, 需要在车间  $m$  台机器上加工, 每个工件有  $m$  道工序, 每个工件在每台机器上仅加工一次。每个用户将加工效率作为目标, 如最大完工时间、加权总拖期、加权总提前期等, 且相互间不公开; 车间也有自身目标, 如车间的总能耗等。

同时假设: 加工工序在生产开始之前确定, 而且在生产过程中不会改变; 任一个工件在任一时刻只能在一台机器上进行加工; 任一时刻一台机器上最多只能加工一个工件; 任一工序只有当其前序工序都加工完成后才能进行加工; 任一工序一旦开始

加工, 不能中断。

若将每个用户和车间均看成不同代理, 共有  $num\_a$  个代理, 第  $a\_i$  个代理拥有工件集  $J_{set}^{a\_i}$  ( $a\_i=1, \dots, num\_a$ ), 将所有调度方案的集合称为调度方案总集  $P_{all} = \bigcup p_a$ , 调度方案  $p_a$  对第  $a\_i$  个代理的目标值用  $o_{a\_i}(p_a)$  表示。

定义 1(Pareto 支配): 多代理调度问题的两个调度方案  $p_a, p_b \in P_{all}, p_a \neq p_b$ , 若  $\forall a\_i=1, 2, \dots, num\_a, o_{a\_i}(p_a) \leq o_{a\_i}(p_b)$  且  $\exists a\_i=1, 2, \dots, num\_a, o_{a\_i}(p_a) < o_{a\_i}(p_b)$ , 则称  $p_a$  支配  $p_b$ , 记为  $p_a \prec p_b$ 。

定义 2(Pareto 最优): 若对于调度方案  $p^* \in P_{all}$ , 在  $P_{all}$  中, 不存在方案  $p'$ , 使得  $p' \prec p^*$ , 则称  $p^*$  为 Pareto 最优方案。

定义 3(Pareto 最优方案集): 所有 Pareto 最优方案组成的集合, 称为 Pareto 最优方案集, 记为  $P_{pareto}$ ,  $P_{pareto} = \{p^* | \neg \exists p \in P_{all} : p \prec p^*\}$

由上述定义可知, 在 Pareto 最优方案集  $P_{pareto}$  中, 不存在仅优化自身效用而不造成其他代理效用恶化的方案。因此, 代理在  $P_{pareto}$  中即是最差的方案也不存在改善的可能, 该最差方案对应的目标函数值可以作为效用函数的计算基准。此外, 由于非支配关系的存在, 代理在  $P_{all}$  中最优方案与  $P_{pareto}$  中的最优方案是一致的。

定义 4(效用函数): 调度方案  $p_a$  对代理  $a\_i$  的效用函数值为

$$u_{a\_i}(p_a) = \frac{\max_{z \in P_{pareto}} o_{a\_i}(p_z) - o_{a\_i}(p_a)}{\max_{z \in P_{pareto}} o_{a\_i}(p_z) - \min_{z \in P_{pareto}} o_{a\_i}(p_z)} \quad (1)$$

由式(1)可见, 采用此效用函数有三方面优势, 其一是将代理目标转化成无量纲量; 其二是对代理目标进行了归一化处理, 消除了代理目标值大小不一的影响; 其三是可以保护代理实际目标值的私有信息。

若以各代理效用函数的纳什积作为社会效益, 以社会效益最大为优化目标, 可建立多代理作业车间调度优化模型

$$\max SW(p) = \prod_{a\_i=1}^{num\_a} u_{a\_i}(p) \quad (2)$$

$$\begin{aligned} \text{s.t.} \quad & st_{S_j^h, j} \geq ct_{S_j^{h-1}, j} \\ & st_{i, H_i^k} \geq ct_{i, H_i^{k-1}} \\ & h=1, 2, \dots, n; j=2, 3, \dots, m \\ & i=1, 2, \dots, n; k=2, 3, \dots, m \end{aligned} \quad (3)$$

式中,  $u_{a-i}(p)$  为调度方案  $p$  对代理  $a-i$  的效用函数值;  $st_{S_j^h,j}$  表示任一机器  $M_j$  上加工某一工序  $O_{S_j^h,j}$  的开工时间;  $ct_{S_j^{h-1},j}$  表示该机器加工紧前工序  $O_{S_j^{h-1},j}$  的完工时间;  $st_{i,H_i^k}$  表示工件  $J_i$  的某一道工序  $O_{i,H_i^k}$  的开工时间;  $ct_{i,H_i^{k-1}}$  表示该工件的前驱工件工序  $O_{i,H_i^{k-1}}$  的完工时间。

本研究中, 车间代理的目标是其总能耗最低<sup>[7-8]</sup>, 简便起见, 设车间总能耗  $TE$  为所有机器的开机能耗  $SE_j$ 、加工能耗  $PE_j$  与待机能耗  $IE_j$  之和<sup>[9-10]</sup>

$$TE = \sum_{j=1}^m (SE_j + PE_j + IE_j) \quad (4)$$

其中机器开机能耗  $SE_j$  可视为常数; 设机器  $M_j$  在加工状态、待机状态下的单位时间能耗分别为  $pp_j$  和  $ip_j$ , 则

$$PE_j = pp_j \times \sum_{i=1}^n t_{ij}$$

式中,  $t_{ij}$  机器  $M_j$  的总加工时间。

$$IE_j = \sum_{h=0}^{n-1} ip_j \times it_h$$

式中,  $it_h$  机器  $M_j$  的总空闲时间。

用户代理目标可设定为, 最大完工时间<sup>[11]</sup>  $C_{max}^{a-i} = \max_{i \in J_{set}^{a-i}} C_i$ , 或加权总拖期<sup>[12]</sup>  $WT^{a-i} = \sum_{i \in J_{set}^{a-i}} \omega_i \times \max\{0, C_i - D_i\}$ , 或加权总提前期<sup>[13]</sup>  $WE^{a-i} = \sum_{i \in J_{set}^{a-i}} \omega_i \times \max\{0, D_i - C_i\}$  等。

## 2 协商调度算法

设计的遗传进化-评分决策两阶段协商调度算法 (Genetic decision based two-stage negotiation scheduling algorithm, GDTNA) 由遗传进化阶段和评分决策阶段组成, 其流程图如图 1 所示。遗传进化阶段, 基于遗传规则<sup>[14-15]</sup>, 随机产生初始父代方案集, 各代理依据各自目标对父代方案集进行遗传操作产生提案  $C_{a-i}$ , 所有代理的提案组成子代方案集  $c = \bigcup C_{a-i}$ , 然后以子代方案集更新父代方案集, 不断重复此过程, 直至达到协商轮次, 输出非支配方案集。评分决策阶段, 基于评分投票规则<sup>[16-17]</sup>, 代理对非支配方案集中各方案评分, 然后从中选择总分最大的方案作为最终的协商结果。该方法通过评

分, 较全面地反映了各方案对代理效用函数的差异, 能够选出社会福利较大的方案。

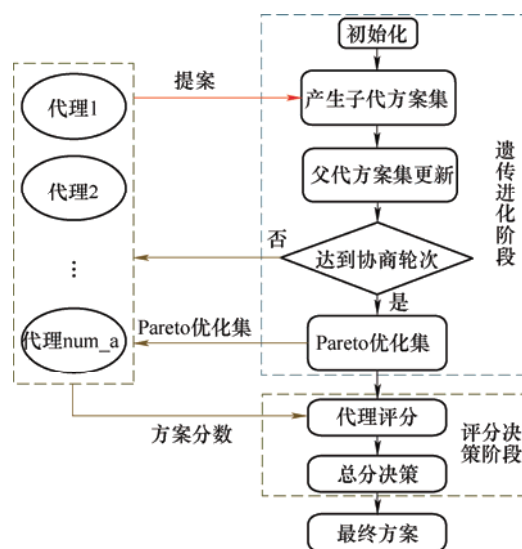


图1 两阶段协商调度算法流程图

### 2.1 编码

作业车间调度遗传算法编码常用的有基于机器排序列表的编码、基于工序优先级的编码等<sup>[18-19]</sup>, 其中, 基于机器排序列表的编码通常基于减少最大完工时间进行调整, 而多代理调度问题的目标不同, 这样做可能会恶化部分代理的目标函数; 基于工序优先级的编码直接按照加工顺序安排各工序, 不会产生冲突, 因此本算法采用基于工序优先级的编码方式。

对于包含  $n$  个工件、 $m$  个机器的多代理作业车间调度问题, 单个编码是长度为  $n \times m$  的数组  $d = \{i \cdots j\}$ , 其中  $i$  表示工件  $J_i$ ,  $i$  第几次出现表示工件  $J_i$  的第几道工序, 对应机器编号相同的工序出现的先后顺序表示工序在该机器上加工的优先顺序, 如  $3 \times 3$  调度问题的一个可能编码个体是  $d = \{1 \ 1 \ 3 \ 2 \ 3 \ 2 \ 2 \ 3 \ 1\}$ ; 此编码方案解码后一定满足约束条件, 且有利于交叉变异等操作。

### 2.2 遗传算子

选择算子采用二元锦标赛的方法, 方案被选中的概率与其目标函数的优良程度呈正相关, 各代理目标维度上优良方案被选中的概率增大。

交叉算子采用两点交叉的方法, 将随机选择的两个父代方案中任意两点间的编码片段进行交换; 为了确保交换后的个体仍是合法个体, 必须对两个交换片段的元素进行逐个检测, 通过个别元素必要的交换, 以保证交换片段中每个元素在原个体中出现的次数完全一致。

变异算子采用位置互换方式, 即随机选取父代

方案中的两个位置,对这两个位置的代码进行交换,以扩大搜索空间,防止方案同质化陷入局部最优。

### 2.3 Pareto 优化算法

本研究中,由于代理目标信息是私有的,无法直接利用代理目标函数值进行 Pareto 优化。为此,我们设计了基于代理排序进行投票的 Pareto 优化方式,即各代理首先按照自身目标对组合方案集进行排序,目标值相同的方案随机排列;随后,各代理按照对应编号方案的排序进行投票,生成 Pareto 优化集;这样可在无需直接披露代理目标信息的情况下实现 Pareto 优化,如算法 1 所示。

算法 1: Pareto 优化。

(1) 代理  $A_{a_i}$  对其当前调度方案集  $P$  按照目标值的升序进行排序,得到排序结果  $R_{a_i} = \{r_1, \dots, r_{p_i}, \dots, r_{num\_p}\}$ , 其中  $num\_p = |P|$ ,  $r_{p_i}$  表示  $P$  中第  $p_i$  个方案  $p_{p_i}$  的序号。

$$R_{p_i}^- = R_{p_i} - R = \begin{bmatrix} r_{1,p_i} - r_{1,1} & \cdots & r_{1,p_i} - r_{1,num\_p} \\ \vdots & & \vdots \\ r_{num\_a,p_i} - r_{num\_a,1} & \cdots & r_{num\_a,p_i} - r_{num\_a,num\_p} \end{bmatrix} \quad (7)$$

③ 如果矩阵  $R_{p_i}^-$  每列均存在小于或等于(且至少 1 列小于)0 的元素,则表示方案  $p_{p_i}$  为 Pareto 非支配方案,将方案  $p_{p_i}$  存入  $P_{pareto}$  中。

④ 令  $p_i$  加 1。

(5) 输出当前 Pareto 优化集  $P_{pareto}$ 。

### 2.4 差值序排除更新算法

差值序排除更新算法之目的是将 Pareto 前沿中与其他方案较相似的方案删除,保持父代方案集的规模和一定的多样性。现有的排除方法先计算两两方案间的欧式距离,选择其中欧式距离最小的两方案并删除其中一个,重复上述过程直至满足父代方案集规模要求。

本研究中,由于代理目标的私有性,无法直接利用代理的目标函数值计算欧式距离,我们提出一种差值序排除算法,其原理是,根据代理对方案间目标差值的排序,间接反映方案间距离的相对大小,并以此为据排除目标差值较小的方案,如算法 2 所示。

算法 2: 差值序排除更新。

(1) 将当前 Pareto 方案集  $F_l = [p_1, \dots, p_{num\_fl}]$  (其中  $num\_fl = |F_l|$ ) 中所有方案组成方案组集  $FP = [(p_1, p_2), \dots, (p_1, p_{num\_fl}), \dots, (p_{num\_fl-1}, p_{num\_fl})]$ , 即  $F_l$  中每个方案都需要与其中另一方案形成一个方

(2) 将各代理的排序结果,按行并列组成如式(5)所示方案排序矩阵

$$R = \begin{bmatrix} r_{1,1} & \cdots & r_{1,num\_p} \\ \vdots & & \vdots \\ r_{num\_a,1} & \cdots & r_{num\_a,num\_p} \end{bmatrix} \quad (5)$$

式中,  $r_{a_i,p_i}$  表示代理  $a_i$  对  $P$  中第  $p_i$  个方案  $p_{p_i}$  的序号。

(3) 令  $p_i=1$ , 初始化 Pareto 优化集  $P_{pareto} = \emptyset$ 。

(4) 当  $p_i < num\_p$ , 重复执行如下步骤。

① 从  $R$  中取出第  $p_i$  列, 并将其列数扩展为  $num\_p$  列, 生成如式(6)

$$R_{p_i} = \begin{bmatrix} r_{1,p_i} & \cdots & r_{1,p_i} \\ \vdots & & \vdots \\ r_{num\_a,p_i} & \cdots & r_{num\_a,p_i} \end{bmatrix} \quad (6)$$

②  $R_{p_i}$  与  $R$  相减, 得到如式(7)

案组。

(2) 各代理  $a_i$  分别计算  $FP$  中两种方案的目标函数差值, 并按照其绝对值升序的方式排序, 得到如式(8)方案组排序矩阵

$$RP_{a_i} = \begin{bmatrix} 0 & \cdots & rp_{1,num\_l} \\ \vdots & & \vdots \\ rp_{num\_l,1} & \cdots & 0 \end{bmatrix} \quad (8)$$

式中,  $rp_{fl-i,fl-j}$  为方案组  $FP$  中方案组  $(p_{fl-i}, p_{fl-j})$  的排序。

(3) 将各代理  $RP_{a_i}$  对应位置的数字平方相加, 得如式(9)估计欧式距离矩阵

$$\widehat{ED} = \begin{bmatrix} 0 & \cdots & \widehat{ed}_{1,num\_l} \\ \vdots & & \vdots \\ \widehat{ed}_{num\_l,1} & \cdots & 0 \end{bmatrix} \quad (9)$$

式中,  $\widehat{ed}_{fl-i,fl-j}$  为  $FP$  中方案组  $(p_{fl-i}, p_{fl-j})$  的估计欧式距离。

(4) 逐行读取  $\widehat{ED}$ , 找出其中值除 0 外最小的值记为  $d_{dmin1,dmin2}$ , 相应的方案组为  $(p_{dmin1}, p_{dmin2})$ 。

(5) 查找矩阵  $\widehat{ED}$  中第  $d_{min1}$  列的除  $\widehat{ed}_{dmin1,dmin2}$  和 0 外的最小值记为  $\widehat{ed}_{dmin1,near1}$ ; 同理得到  $\widehat{ed}_{dmin2,near2}$ 。

(6) 如果  $\widehat{ed}_{dmin1, near1} > \widehat{ed}_{dmin2, near2}$ , 表示方案  $p_{dmin1}$  距离其他方案更远, 即该方案多样性更好, 因此将  $p_{dmin2}$  从  $F_l$  中删除; 反之, 则将  $p_{dmin1}$  从  $F_l$  中删除。

(7) 若  $|F_l| > num\_p$ , 转步骤(1); 否则, 结束。

## 2.5 父代方案集更新算法

以子代方案集和当前父代方案集的集合为基础, 通过 Pareto 优化和差值序排除更新算法, 获得新一代父代方案集, 如算法 3 所示。

算法 3: 父代方案集更新。

(1) 子代方案集  $C$  与当前父代方案集  $P_g$  合并生成组合方案集  $PC$ , 代理  $A_{a_i}$  根据预设的目标函数, 计算方案集  $PC$  中所有方案的目标值。

(2) 初始化方案集  $T_g = \emptyset$ , 并令  $l=0$ 。

(3) 调用“Pareto 优化算法”;  $F_l = P_{Pareto}$ 。

(4)  $T_g = T_g \cup F_l$ , 若  $|T_g| + |F_l| < num\_p$ , 将  $F_l$  对应方案从  $PC$  中删除, 令  $l$  加 1, 返回步骤(3); 否则, 执行步骤(5)。

(5) 调用“差值序排除更新算法”。

(6) 如果  $|T_g| + |F_l| = num\_p$ ,  $P_{g+1} = T_g \cup F_l$ , 输出  $P_{g+1}$ 。

## 2.6 评分决策算法

基于评分投票规则<sup>[20-22]</sup>设计决策算法, 包括代理评分和求积决策两个步骤。代理评分采用基于效用函数值的线性评分算法, 即代理将效用函数值直接乘以满分值  $M$ , 此方法能够较准确反映代理对方案的偏好程度。在求积决策步骤, 将各代理评分直接相乘作为总分, 此方式与纳什积社会福利计算方式相同, 二者成正比, 总分较大等同于社会福利较大, 因此选择总分最大的方案也就是选择社会福利较优的方案作为最终协商结果, 如算法 4 所示。

算法 4: 评分决策。

(1) 各代理分别计算 Pareto 优化集  $P_{Pareto}$  中各方案的目标函数值。

(2) 各代理按式(1)计算  $P_{Pareto}$  中各方案的效用函数值。

(3) 各代理按式  $c_{a-i}(p_i) = M \times \hat{u}_{a-i}(p_i)$  计算  $P_{Pareto}$  各方案评分。

(4) 按式  $cp(p_i) = \prod_{a-i=1}^{num\_a} c_{a-i}(p_i), p_i \in P_{Pareto}$  计算  $P_{Pareto}$  中各方案的总分。

(5) 选择  $P_{Pareto}$  中评分最大方案输出  $p_{output} =$

$$\underset{p_i \in P_{Pareto}}{\operatorname{argmax}}(cp(p_i))。$$

## 3 仿真研究

为了评估所提遗传进化-评分决策两阶段协商调度算法(GDTNA)的性能, 基于作业车间调度标准问题 Abz、Ft、La、Swv、Ta 和 Yn 类等 32 个基准问题<sup>[14, 23-25]</sup>, 分别随机生成代理规模为 3、4、5 的问题实例各 3 个, 共计 288 个多代理作业车间调度问题算例。

仿真研究中使用的对比算法分别为随机搜索进化-决策两阶段协商调度算法(RDTNA)<sup>[2]</sup>、基于模拟退火的协商调度算法(SANA)<sup>[1]</sup>和基于遗传算法的协商调度算法(GANA)<sup>[3]</sup>。

仿真研究中, 各用户代理的目标按随机方式确定, 即对每个用户代理产生一个 1~3 之间的随机整数, 由随机整数值决定各用户代理的目标, 即 1=最大完工时间最小、2=加权总拖期最小、3=加权总提前期最小; 车间代理的目标为总能耗最低。

仿真算法采用 Python 编程实现, 计算机具有 2.8 GHz CPU 和 8GB RAM, 各算法的参数设置如表 1 所示。

表 1 调度算法参数

参数类型	GDTNA	RDTNA	SANA	GANA
协商轮次	2 000	2 000	2 000	2 000
子代方案数目	$100 \times num\_a$	$100 \times num\_a$	$100 \times num\_a$	$100 \times num\_a$
父代方案数目	$100 \times num\_a$			$100 \times num\_a$
变异概率	0.05			0.05

采用平均性能相对指标(PR), 作为对比标准

$$PR(x) = \frac{1}{num\_ins} \sum_{i=1}^{num\_ins} \frac{V_i(x)}{\max_{x_j \in Ae} (V_i(x_j))} \quad (10)$$

式中,  $num\_ins$  代表同一基准问题相同代理数目的实例数,  $Ae$  为实验  $e$  中进行对比的所有算法,  $V_i(x)$  为使用算法  $x$  在问题实例  $i$  上产生调度方案的性能指标值。

本研究中, 各代理性能指标, 如车间总能耗、最大完工时间、加权总拖期、加权总提前期等均希望取最小值, 因此从式(10)可见,  $PR$  值是无量纲量, 且其值越小, 说明调度算法性能越好。

3 代理、4 代理和 5 代理的仿真结果如表 2 所示, 变化趋势分别如图 2~4 所示。

表 2 协商调度算法 PR 指标值

算例	3 代理				4 代理				5 代理			
	GDTNA	RDTNA	SANA	GANa	GDTNA	RDTNA	SANA	GANa	GDTNA	RDTNA	SANA	GANa
Abz5	<b>0.033</b>	0.617	0.719	1.000	<b>0.231</b>	0.461	1.000	0.748	<b>0.202</b>	0.505	0.817	0.508
Abz6	<b>0.084</b>	0.597	0.952	0.750	<b>0.102</b>	0.734	0.484	0.951	<b>0.010</b>	0.596	0.742	0.722
Abz7	<b>0.000</b>	0.560	1.000	0.598	<b>0.038</b>	0.834	0.986	0.694	<b>0.190</b>	0.698	0.932	0.603
Abz8	<b>0.064</b>	0.693	0.426	0.981	<b>0.247</b>	0.758	1.000	0.628	<b>0.246</b>	0.703	1.000	0.612
Abz9	<b>0.047</b>	0.805	0.748	0.898	<b>0.045</b>	0.617	0.834	0.873	<b>0.184</b>	0.694	0.911	0.800
Ft06	<b>0.495</b>	0.607	0.973	0.827	<b>0.010</b>	0.282	1.000	0.578	0.309	<b>0.188</b>	0.947	0.792
Ft10	<b>0.024</b>	0.915	0.799	0.791	<b>0.073</b>	0.839	0.627	0.758	<b>0.234</b>	0.716	0.827	0.328
Ft20	<b>0.181</b>	0.687	0.303	0.748	<b>0.230</b>	0.910	0.727	0.591	<b>0.632</b>	0.817	0.748	0.772
La01	<b>0.062</b>	0.820	0.701	0.215	<b>0.085</b>	0.682	0.786	0.255	<b>0.307</b>	0.567	0.767	0.903
La06	<b>0.290</b>	1.000	0.727	0.798	<b>0.242</b>	0.877	0.927	0.814	0.493	0.722	1.000	<b>0.304</b>
La11	<b>0.126</b>	0.640	0.320	0.880	<b>0.335</b>	0.859	0.824	0.898	<b>0.614</b>	0.631	0.841	0.645
La16	<b>0.000</b>	0.705	0.621	0.632	<b>0.194</b>	0.695	0.911	0.827	<b>0.324</b>	0.786	0.839	0.871
La21	<b>0.059</b>	1.000	0.178	0.865	<b>0.000</b>	0.870	1.000	0.739	<b>0.231</b>	0.927	0.753	0.643
La26	<b>0.025</b>	0.975	0.453	0.977	<b>0.036</b>	0.827	0.752	0.798	<b>0.000</b>	0.920	0.779	0.864
La31	<b>0.110</b>	0.818	0.507	0.833	<b>0.087</b>	0.974	0.538	0.796	<b>0.263</b>	0.667	0.906	0.765
La36	<b>0.001</b>	0.942	0.427	0.768	<b>0.020</b>	0.653	0.998	0.747	<b>0.018</b>	0.817	0.756	0.868
Swv05	<b>0.112</b>	1.000	0.544	0.691	<b>0.308</b>	1.000	0.565	0.866	<b>0.592</b>	0.873	0.815	0.612
Swv10	<b>0.159</b>	0.919	0.493	0.876	<b>0.182</b>	0.843	0.819	0.678	<b>0.291</b>	0.932	0.801	0.897
Swv15	<b>0.122</b>	0.958	0.535	0.958	<b>0.554</b>	0.992	0.583	0.951	<b>0.409</b>	1.000	0.499	0.966
Swv20	<b>0.218</b>	0.911	0.424	0.856	<b>0.126</b>	0.789	0.795	0.872	<b>0.151</b>	0.824	0.746	0.693
Ta10	<b>0.105</b>	0.927	0.525	0.892	<b>0.058</b>	0.878	0.557	0.907	<b>0.124</b>	0.509	0.743	0.790
Ta20	<b>0.024</b>	0.472	0.746	0.998	<b>0.182</b>	0.826	0.728	0.813	<b>0.169</b>	0.757	1.000	0.706
Ta30	<b>0.113</b>	1.000	0.238	0.969	<b>0.050</b>	0.835	0.729	0.839	<b>0.234</b>	0.618	0.913	0.459
Ta40	<b>0.088</b>	1.000	0.463	0.962	<b>0.051</b>	0.797	0.838	1.000	<b>0.097</b>	0.575	0.999	0.636
Ta50	<b>0.073</b>	0.789	0.418	0.935	<b>0.047</b>	0.742	0.714	0.729	<b>0.095</b>	0.701	0.968	0.779
Ta60	<b>0.231</b>	0.942	0.603	0.999	<b>0.198</b>	0.813	0.823	0.926	<b>0.117</b>	0.925	0.544	0.961
Ta70	<b>0.105</b>	0.988	0.689	0.939	<b>0.143</b>	0.927	0.690	0.822	<b>0.182</b>	0.622	0.893	0.680
Ta80	<b>0.092</b>	0.679	1.000	0.706	<b>0.106</b>	0.809	0.761	0.914	<b>0.080</b>	0.801	0.765	0.791
Yn1	<b>0.166</b>	0.833	0.312	0.991	<b>0.088</b>	0.578	0.767	0.928	<b>0.086</b>	0.825	0.710	0.875
Yn2	<b>0.001</b>	0.761	0.942	0.822	<b>0.051</b>	0.674	0.795	0.786	<b>0.196</b>	0.786	0.684	0.657
Yn3	<b>0.174</b>	0.914	0.413	0.988	<b>0.014</b>	0.841	0.775	0.739	<b>0.287</b>	0.684	0.743	0.875
Yn4	<b>0.150</b>	0.830	0.760	0.701	<b>0.155</b>	0.632	0.897	0.688	<b>0.044</b>	0.849	0.731	0.995

从表 2 可见, 不管是 3 代理、4 代理还是 5 代理, GDTNA 的  $PR$  值在所有仿真算例中均是最小的(表 2 中加粗数据, 5 代理 Ft06 和 La06 算例例外); 此外, 在不区分代理数的情况下, GDTNA、RDTNA、SANA 和 GANA 的  $PR$  均值和方差分别为 0.159、

0.775、0.732、0.785 和 0.138、0.159、0.198、0.163, 即 GDTNA 的  $PR$  均值仅为其余三种算法的 20% 左右, 而方差大小无实质性差异、且均较小, 说明 4 种算法  $PR$  值相对于其均值的离散度较小; 因此, 本文所提算法是所有参与比较算法中性能最好的。



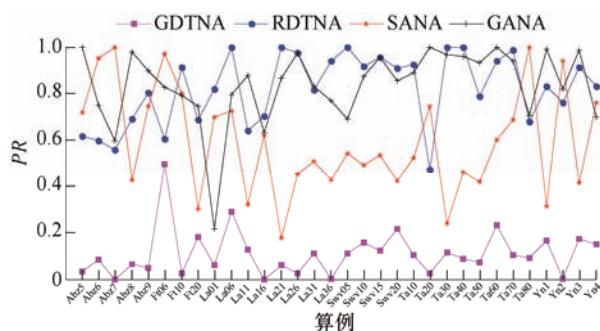


图2 协商调度算法整体性能PR值分布(3代理)

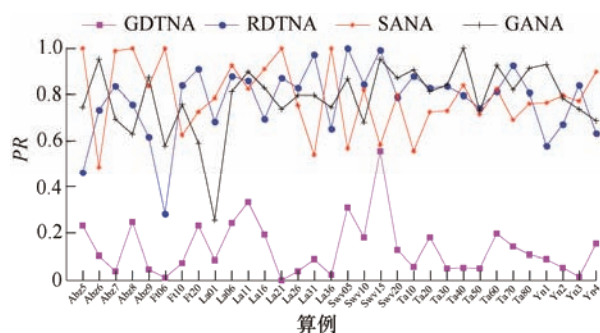


图3 协商调度算法整体性能PR值分布(4代理)

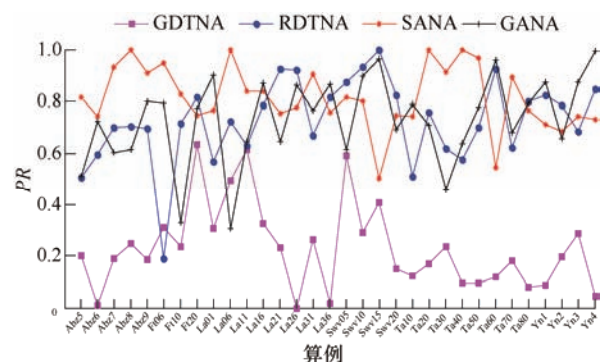


图4 协商调度算法整体性能PR值分布(5代理)

从图2~4可见,不管是3代理、4代理还是5代理,GDTNA的PR值一般均小于其余三种算法,且各种算法PR值变动幅度都不大,因此,GDTNA的整体性能优于其余三种算法,GDTNA在大部分算例上都能取得更高的社会福利。

但是,从图2~4和表2也可见,GDTNA的PR均值从3代理的0.110上升为5代理的0.231,而RDTNA的PR均值从3代理的0.822下降为5代理的0.726,这表明GDTNA的整体性能随代理数增加相对下降,而RDTNA的整体性能随代理数增加相对提升,其原因可能为,GDTNA遗传进化阶段算法的收敛性随着代理数增多而有所降低,导致其整体性能下降,这也是今后的研究方向之一。

## 4 结论

本文针对代理目标为私有信息的多代理作业车间调度问题,基于遗传进化机制设计了一种多代理作业车间遗传进化-评分决策两阶段协商优化调度决策算法,通过用户直接参与车间调度方案的决策过程,可以在不披露用户私有目标信息的前提下,实现协调用户间的竞争需求,并产生车间与用户整体社会福利较优的调度方案。仿真研究基于8类32个标准作业车间调度问题,构建了288个多代理作业车间调度问题算例,仿真研究结果表明,文中所提遗传进化-评分决策两阶段协商优化调度策略可行;与现有多代理协商调度算法相比,遗传进化-评分决策两阶段协商调度算法可以产生社会福利更高的调度方案。因此,本研究在不披露用户私有目标信息前提下,解决了已有协商调度算法存在收敛性不足和产生的调度方案社会福利较低的问题;研究结果对含有私有目标信息的多代理作业车间调度具有较好应用价值。

## 参考文献

- [1] LANG F, FINK A, BRANDT T. Design of automated negotiation mechanisms for decentralized heterogeneous machine scheduling[J]. *European Journal of Operational Research*, 2016, 248(1): 192-203.
- [2] HOMBERGER J, FINK A. Generic negotiation mechanisms with side payments – design, analysis and application for decentralized resource-constrained multi-project scheduling problems[J]. *European Journal of Operational Research*, 2017, 261(3): 1001-1012.
- [3] HOMBERGER J. A  $(\mu, \lambda)$ -coordination mechanism for agent-based multi-project scheduling[J]. *OR Spectrum*, 2012, 34(1): 107-132.
- [4] FINK A, HOMBERGER J. An ant-based coordination mechanism for resource-constrained project scheduling with multiple agents and cash flow objectives[J]. *Flexible Services and Manufacturing Journal*, 2013, 25(1-2): 94-121.
- [5] LANG F, FINK A. Learning from the metaheuristics: Protocols for automated negotiations[J]. *Group Decision and Negotiation*, 2015, 24(2): 299-332.
- [6] 张洁, 盛夏, 张朋, 等. 面向制造过程数据的两阶段无监督特征选择方法[J]. *机械工程学报*, 2019, 55(17): 133-144.  
ZHANG Jie, SHENG Xia, ZHANG Peng, et al. Two-stage unsupervised feature selection method oriented to

- manufacturing procedural data[J]. *Journal of Mechanical Engineering*, 2019, 55(17): 133-144.
- [7] EBRAHIMI A, JEON H W, LEE S, et al. Minimizing total energy cost and tardiness penalty for a scheduling-layout problem in a flexible job shop system: A comparison of four metaheuristic algorithms[J]. *Computers & Industrial Engineering*, 2020, 141: 106295.
- [8] 刘琼, 梅侦. 面向低碳的工艺规划与车间调度集成优化[J]. *机械工程学报*, 2017, 53(11): 164-174.
- LIU Qiong, MEI Zhen. Integrated optimization of process planning and shop scheduling for reducing manufacturing carbon emissions[J]. *Journal of Mechanical Engineering*, 2017, 53(11): 164-174.
- [9] 李聪波, 沈欢, 李玲玲, 等. 面向能耗的多工艺路线柔性作业车间分批优化调度模型[J]. *机械工程学报*, 2017, 53(5): 12-23
- LI Congbo, SHEN Huan, LI Lingling, et al. A batch splitting flexible job shop scheduling model for energy saving under alternative process plans[J]. *Journal of Mechanical Engineering*, 2017, 53(5): 12-23.
- [10] 吴秀丽, 孙阳君. 机器多转速的柔性作业车间绿色调度问题[J]. *计算机集成制造系统*, 2018, 24(4): 862-875
- WU Xiuli, SUN Yangjun. Flexible job shop green scheduling problem with multi-speed machine[J]. *Computer Integrated Manufacturing Systems*, 2018, 24(4): 862-875.
- [11] 吴晓强, 田松龄, 郑淑贤, 等. 面向开放式结构产品的智能制造车间实时调度方法[J]. *计算机集成制造系统*, 2020, 26(10): 2703-2713
- WU Xiaoqiang, TIAN Songling, ZHENG Shuxian, et al. Real-time scheduling method of intelligent manufacturing shop floor based on multi-agent[J]. *Computer Integrated Manufacturing Systems*, 2020, 26(10): 2703-2713.
- [12] 雷德明, 苏斌. 基于多班教学优化的多目标分布式混合流水车间调度[J]. *控制与决策*, 2021, 36(2): 303-313
- LEI Deming, SU Bin. Multi-class teaching-learning-based optimization for multi-objective distributed hybrid flow shop scheduling[J]. *Control and Decision*, 2021, 36(2): 303-313.
- [13] WU Zigao, SUN Shudong. Risk cost estimation of job shop scheduling with random machine breakdowns[C]// 11th CIRP Conference on Industrial Product-Service Systems, Zhuhai, 29-31 May 2019, *Procedia CIRP* 2019, 83: 404-409.
- [14] LI Xinyu, GAO Liang. An effective hybrid genetic algorithm and tabu search for flexible jobshop scheduling problem[J]. *International Journal of Production Economics*, 2016, 174: 93-110.
- [15] 周明, 孙树栋. 遗传算法原理及应用[M]. 北京: 国防工业出版社, 1999.
- ZHOU Min, SUN Shudong. Genetic algorithm: Theory and applications[M]. Beijing: National Defense Industry Press, 1999.
- [16] PIVATO M. Formal utilitarianism and range voting[J]. *Mathematical Social Sciences*, 2014, 67: 50-56.
- [17] WIBOWO Y, MAHARDIKA N S, AFRIZAL R, et al. Design of the expert system for Edamamegrading using forward chaining method[J]. *Sustainable Environment Agricultural Science*, 2020, 4(1): 26-37.
- [18] GONCALVES J F, DE MAGALHAES MENDES J J, RESENDE M G C. A hybrid genetic algorithm for the job shop scheduling problem[J]. *European Journal of Operational Research*, 2005, 167(1): 77-95.
- [19] TANG D, DAI M. Energy-efficient approach to minimizing the energy consumption in an extended job-shop scheduling problem[J]. *Chinese Journal of Mechanical Engineering*, 2015, 28: 1048-1055.
- [20] HILLINGER C. The case for utilitarian voting[J]. *Discussion Papers in Economics*, 2005, 23(3): 295-321.
- [21] SEN A. Majority decision and Condorcet winners[J]. *Social Choice and Welfare*, 2020, 54(1): 1-7.
- [22] NITISIRI K, GEN M, OHWADA H. A parallel multi-objective genetic algorithm with learning-based mutation for railway scheduling[J]. *Computers & Industrial Engineering*, 2019, 130: 381-394.
- [23] WANG J Q, FAN G Q, ZHANG Y Q, et al. Two-agent scheduling on a single parallel-batching machine with equal processing time and non-identical job sizes[J]. *European Journal of Operational Research*, 2017, 258(2): 478-490.
- [24] ZHANG Y F, ZHANG G, WANG J Q, et al. Real-time information capturing and integration framework of the internet of manufacturing things[J]. *International Journal of Computer Integrated Manufacturing*, 2015, 28(8): 811-822.
- [25] NIU G G, SUN S D, LAFON P, et al. A decomposition approach to job-shop scheduling problem with discretely controllable processing times[J]. *Science China: Technological Sciences*, 2011, 54(5): 1240-1248.

作者简介: 孙树栋(通信作者), 男, 1963 年出生, 教授, 博士研究生导师。主要研究方向为数字化制造技术。

E-mail: sdsun@nwpu.edu.cn

周新民, 男, 1969 年生, 博士研究生。主要研究方向为制造业信息化技术。

E-mail: siem@nwpu.edu.cn

常昇博, 男, 1998 年生, 硕士研究生。主要研究方向为制造系统建模与优化技术。

E-mail: shenbo1998@qq.com