

文章编号: 1001-0920(2011)12-1761-07

## 求解双资源约束车间调度问题的继承式双目标遗传算法

李兢尧, 孙树栋, 黄媛, 牛刚刚

(西北工业大学 a. 系统集成与工程管理研究所, b. 现代设计与集成制造教育部重点实验室, 西安 710072)

**摘要:** 针对双资源约束作业车间调度的双目标优化问题, 提出一种继承式遗传算法, 通过分支种群继承父辈种群的进化经验. 该算法面向双资源约束特点, 采用4维染色体编码方式, 基于时间窗口比较实现活动化调度, 通过资源进化算子提高算法全局搜索能力; 基于个体 Pareto 指数的锦标赛选择策略, 有效削弱了染色体 Pareto 排序级别对个体存活概率的影响以保持群体多样性, 并利用精英保留策略提高了解的收敛性. 仿真实验与分析结果表明了所提算法具有优良性能.

**关键词:** 双资源约束; 继承式遗传算法; 时间窗口比较; Pareto 指数

中图分类号: TP18

文献标识码: A

## Double-objective inherited genetic algorithm for dual resource constrained job shop

LI Jing-yao, SUN Shu-dong, HUANG Yuan, NIU Gang-gang

(a. System Integration and Engineering Management Institution, b. Key Laboratory of Contemporary Design and Integrated Manufacturing Technology of Ministry of Education, Northwestern Polytechnical University, Xi'an 710072, China.

Correspondent: LI Jing-yao, E-mail: ljyao.6106@163.com)

**Abstract:** To solve the double-objective optimal of dual resource constrained job shop scheduling problem, an inherited genetic algorithm is proposed, in which the evolutionary experience of parent population is inherited by the means of branch population generation with pheromones to accelerate the convergence rate. Meanwhile, by using the four-dimensional chromosome coding method, based on comparison among time windows, the activable decoding algorithm is utilized with reference to the character of dual resource constrained to improve the overall searching ability. During the evolution process, the championship selection strategy based on Pareto index weakens the impact of the Pareto level of chromosomes obviously to keep the community diversity. The reliable convergence of algorithm is guaranteed by using elitist preservation strategy. The simulation experiment and statistical analysis on extant and random example show that the proposed method has good performance.

**Key words:** dual resource constrained; inherited GA; time windows comparison; Pareto index

### 1 引言

1967年 Nelson<sup>[1]</sup>突破了车间调度问题(JSP)中传统的狭义资源定义, 由单纯的机器设备外延到人力资源等辅助加工资源, 并将这种同时考虑设备、工人约束的JSP定义为双资源约束(DRC)JSP. 它是基于柔性作业车间调度问题(FJSP). 面向实际生产环境的进一步扩展, 每道工序可由多个工人操作多台设备加工, 且不同资源组合的加工效果不同. 此后, 国内外诸多学者相继对此类问题展开研究, 主要集中在 when, where, who 等工人转移规则的影响<sup>[2-4]</sup>及启发式调度

算法<sup>[5]</sup>两方面, 但多为基于时间指标的单目标优化, 导致加工能力较强且成本较高的资源超负荷运转, 时间指标最优的同时恶化了成本指标, 与利益最大化的企业最终目标相悖. 因此, 有效协调时间-成本间矛盾关系的双目标优化研究, 在双资源约束环境中具有重要的理论意义和应用价值.

由于指标间耦合关系, 不存在一个同时多指标最优的调度方案, 多目标调度优化只能得到由一组相对最优解构成的 Pareto 解集. 目前多目标进化算法主要有两种研究思路: 1) 首先通过群体分割进化<sup>[6]</sup>或随

收稿日期: 2010-07-09; 修回日期: 2010-12-08.

基金项目: 国家863/CIMS计划项目(2007AA04Z187); 国家自然科学基金项目(50705076, 50705077).

作者简介: 李兢尧(1984-), 男, 博士生, 从事智能调度优化的研究; 孙树栋(1963-), 男, 教授, 博士生导师, 从事制造单元调度优化、机器人控制等研究.

机权值和<sup>[7]</sup>等方式将多目标问题转化为基于不同指标的单一目标问题,然后借助成熟的单一目标进化算法分别求解,此方法实现简单,算法开发量小,但仅能得到部分极值点,难以收敛到 Pareto 最优沿面的非凸区域;2)对并行进化算法每次优化周期内获得的一系列可行解进行 Pareto 分级排序,基于适应值共享策略鼓励 Pareto 等级较高且较稀疏个体存活,此方法无需非凸性假设且全局搜索能力较强,所得 Pareto 解集分布更均匀,是目前多目标进化的研究热点。

以种群为优化单位实现多向并行搜索的遗传算法(GA)用于求解多目标进化问题时能够兼顾解的收敛性及多样性,因此被诸多学者选为基于 Pareto 排序多目标进化算法的主体框架,如:多目标遗传算法(MOGA)<sup>[8]</sup>,基于非支配排序的二代遗传算法(NSGAI)<sup>[9]</sup>, Pareto 包络进化策略(PAES)<sup>[10]</sup>,强度 Pareto 进化算法(SPEA)<sup>[11]</sup>等。但这种由单一种群出发不断逼近最优种群的进化模式仅遍历有限解空间且过分依赖初始群体。Aiex 等人<sup>[12]</sup>曾提出一种贪婪随机自适应搜索算法(GRASP),在每次迭代后初始化群体,虽然在群体多样性方面获得了良好效果,但却未能充分利用当前已搜索的信息,收敛性较差。

本文引入蚁群算法(ACO)中基于信息素累积传递蚂蚁寻优经验实现群体收敛的思想,提出一种基于 Pareto 指数的双目标继承式遗传算法(DOIGA),并用于 DRCJSP 双目标优化求解。

## 2 继承式 GA 双目标调度算法

DOIGA 算法将每个工序的多道工艺视为多条分

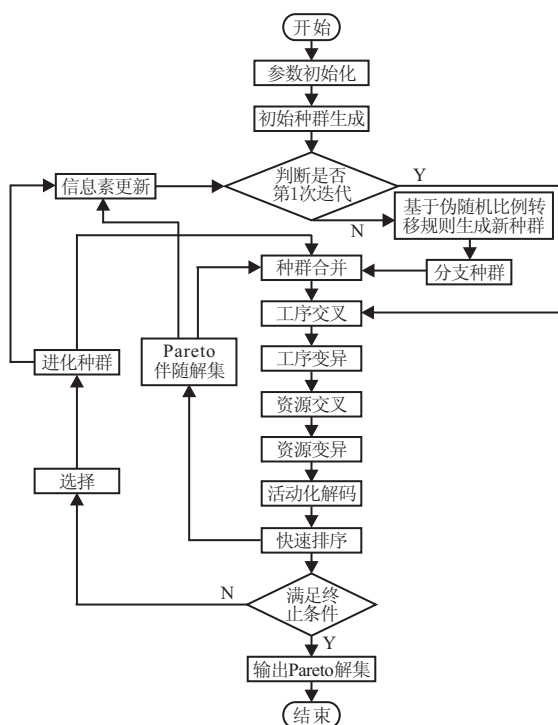


图 1 DOIGA 算法结构图

支路径,每次迭代前利用基于路径信息素及启发式信息的伪随机比例状态转移规则生成分支种群,既增强了进化种群的多样性,扩大解空间的搜索范围,又以信息素为载体继承上一代的“生存经验”,加快了算法收敛。此外,在进化过程中,算法兼顾了工序调度及资源分配的调整,实现了二者的同步优化;基于染色体的双目标支配关系比较获取 Pareto 指数,作为个体间优劣关系的评判标准;以伴随解集记录每次迭代的 Pareto 解集,并参与迭代进化以保证算法收敛。最后,基于 Pareto 解集及进化种群进行信息素更新,保存先进的进化经验。DOIGA 算法结构如图 1 所示。

### 2.1 基于信息素的经验继承

GA 通过对初始种群的各种进化操作以及优胜劣汰机制实现算法收敛,但进化过程中父代与子代之间缺乏信息交流,较强的全局搜索能力是以缓慢的收敛速度为代价。DOIGA 引入信息素概念,反映各工艺对调度目标的影响,并利用改进伪随机比例状态转移规则生成染色体分支种群,以信息素为载体继承父辈的进化经验,引导群体的进化方向。

一般 ACO 中蚁群仅通过信息素浓度和路径启发式信息选择路径,但 Dussutouretal<sup>[13]</sup>通过观察蚁群在不同宽度的对称双分支桥上的移动过程,揭示了另一种影响因素——蚂蚁流量。实验发现,当蚂蚁流量较小时,所有蚂蚁最终选择同一分支;当蚁群高度拥挤时,两条分支的蚂蚁数量相当,如图 2(a)所示。这一现象是由蚂蚁处理堵塞的方法所决定的,即蚂蚁在路径交叉口相遇时,先出发蚂蚁会将后来的蚂蚁挤到“辅路”上去,如图 2(b)所示。可见蚁群算法实质并不是一种完全并行的搜索算法,先出发蚂蚁具有路径优先选择权。

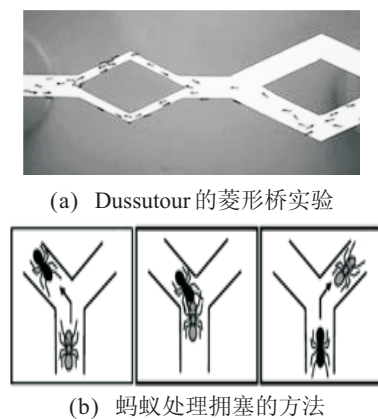


图 2 蚁群算法示意图

综上所述,基于蚂蚁流量的改进伪随机比例路径转移规则如下:

$$P_{P_x} = \begin{cases} 1, & P \leq q \wedge \arg \max_{P_x \in S_0} \{(\tau_{P_x}^{O_1} \times \tau_{P_x}^{O_2})^\alpha \times \eta_{P_x}^\beta \times n_{P_x}^\gamma\}; \\ \frac{(\tau_{P_x}^{O_1} \times \tau_{P_x}^{O_2})^\alpha \times \eta_{P_x}^\beta \times n_{P_x}^\gamma}{\sum_{P_x \in S_0} ((\tau_{P_x}^{O_1} \times \tau_{P_x}^{O_2})^\alpha \times \eta_{P_x}^\beta \times n_{P_x}^\gamma)}, & P > q. \end{cases} \quad (1)$$

其中:  $P_{P_x}$  为备选工艺集合  $S_0$  中工艺  $P_x$  被选择概率; 转移概率  $q$  是平衡算法探索、利用性能的重要参数, 蚂蚁以概率  $q$  选择当前已开发的最优移动方式, 并以概率  $(1 - q)$  有偏向性地探索新路径;  $\tau_{P_x}^{O_1}, \tau_{P_x}^{O_2}$  分别为每道工序针对成本和时间指标的信息素;  $\eta_{P_x}$  为基于启发式规则生成的启发式信息;  $n_{P_x}$  为路径流量, 即该工艺被选择的次数。

### 2.2 编码及活动化解码

Cheng<sup>[14]</sup>总结了 9 种求解 JSP 时 GA 所用编码方式, 其中基于工件的整数编码由于实现简单且能保证调度可行性而广泛使用, 但用于 DRCJSP 求解时染色体仅包含显性工件与隐形工序信息, 资源只能在解码阶段通过启发式规则配置<sup>[5]</sup>, 染色体与调度方案形成单对多映射, 编码的唯一性被破坏. 对此, DOIGA 采用 4 维编码方式, 染色体  $R_c$  的每个基因  $G_i$  由 4 个元素  $G_i^p (p = 1, 2, 3, 4)$  组成, 分别表示工件、工序、设备和工人, 如图 3 所示. 第 1 个基因表示由工人 3 操作设备 1 加工工件 1 的第 2 道工序. 染色体与调度方案一一对应, 表现直观且解码方便.

工件	...	1	1	3	2	...	1	2	...
工序	...	2	3	3	4	...	4	5	...
设备	...	1	5	2	3	...	2	3	...
工人	...	3	3	2	3	...	3	4	...

图 3 染色体 4 维编码方式

根据 Venn 调度关系图, 最优调度一定是活动调度, 调度算法若只在活动调度集合内搜索最优解, 则可大幅提高算法搜索效率. 针对双资源约束特点, 本文提出一种基于时间窗口比较的活动化解码算法, 每道工序选择资源组合可用时间窗口集合中满足其工艺约束的最早时间窗口进行加工, 根据文献 [15] 的成本计算公式, 此方法能实现时间-成本的双目标活动化调度解码. 首先给出关于时间窗口比较的重要定义:

**定义 1** 时间窗口相交. 当时间窗口  $A[t_A^S, t_A^E]$  与时间窗口  $B[t_B^S, t_B^E]$  满足  $t_A^S < t_B^E$  且  $t_A^E > t_B^S$  时, 称  $A$  与  $B$  存在交集  $C[\max(t_A^S, t_B^S), \min(t_A^E, t_B^E)]$ , 记为  $A \cap B = C$ .

**定义 2** 时间窗口包含. 当时间窗口  $A$  与时间窗口  $B$  满足  $A \cap B = A$  时, 称  $B$  包含  $A$ , 记为  $B \supset A$ .

**定义 3** 时间窗口集合相交. 现有两个时间集合

$$C_1^T = \bigcup_{i=1}^{n_i} A_i, C_2^T = \bigcup_{j=1}^{n_j} B_j,$$

$A_i (i = 1, 2, \dots, n_i)$  和  $B_j (j = 1, 2, \dots, n_j)$  均为散布在时间区域  $[0, +\infty)$  内的时间窗口, 且  $A_i$  之间,  $B_j$  之间互不相交. 当

$$C_3^T = \left\{ \bigcup_{i=1}^{n_i} C_i \mid C_i = \bigcup_{j=1}^{n_j} (A_i \cap B_j) = A_i \cap C_2^T \right\} \neq \emptyset$$

时, 称时间窗口集合  $C_1^T$  与  $C_2^T$  相交, 记为  $C_3^T = C_1^T \cap C_2^T$ .

基于上述定义, 活动化解码算法步骤如下:

1) 以集合  $C_P$  记录各工件当前工序的完成时间, 以集合  $C_W, C_M$  分别记录各个工人、设备的可用时间窗口 (初始均为  $[0, +\infty)$ ), 依基因顺序遍历染色体所有基因;

2) 对于基因  $G_i$ , 从  $C_P$  获得该工件上一道工序的完成时间为  $t_{(G_i^2-1)}^e$  (若  $G_i^2 = 1$ , 则  $t_{(G_i^2-1)}^e = 0$ ), 已知  $G_i^2$  的加工时间为  $t_{G_i^2}^p$ ;

3) 从  $C_W, C_M$  分别获得  $G_i^3$  与  $G_i^4$  的可用时间窗口集合并依定义 3 求得二者交集  $C_R^T$ ;

4)  $A = \{\min t_A^S \mid \exists A[t_A^S, t_A^E] \in C_R^T \wedge A \supset [\max(t_A^S, t_{(G_i^2-1)}^e), \max(t_A^S, t_{(G_i^2-1)}^e) + t_{G_i^2}^p]\}$ ,  $G_i^2$  的加工窗口为  $[\max(t_A^S, t_{(G_i^2-1)}^e), \max(t_A^S, t_{(G_i^2-1)}^e) + t_{G_i^2}^p]$ ;

5) 分别更新集合  $C_P, C_W$  和  $C_M$ , 继续下一个基因调度, 直至染色体遍历完成.

### 2.3 交叉与变异

GA 算法中的两种进化算子: 交叉和变异, 基于适应值函数设计, 分别通过染色体基因的交换及突变操作实现染色体的优秀基因传递及种群优胜劣汰, 前者执行随机搜索试图跳出局部最优, 后者执行局部搜索寻找最优解, 二者结合使遗传进化兼具广度、深度搜索能力, 在进化搜索中具备同等重要地位. 但一般 GA 算法为避免基因突变使精英个体的优秀基因丧失, 变异概率一般远小于交叉概率, 隐性削弱了 GA 的局部搜索能力; 另一方面, 针对双目标优化问题, 种群的多样性与收敛性需同时兼顾. 因此 DOIGA 算法采取全交叉变异进化方式, 即交叉概率和变异概率均为 1, 进化阶段不淘汰个体, 所有父、子种群均进入 Pareto 排序选择阶段, 以增强种群进化的多样性.

为保证算法的全局搜索能力, DOIGA 算法在工序进化阶段基于多种进化算子共同构成的复合算子集增强搜索行为的多样性. 工序交叉算子集包括顺序交叉、线性顺序交叉和子串交换等; 工序变异算子集则有互反变异与插入变异, 在相应进化阶段由染色体

独立选择一种执行. 另外, 为避免确定性较强的启发式资源配置策略导致的局部最优, 除工序进化算子外, DOIGA 额外引入 3 类资源进化算子以实现工序调度与双资源配置的同步进化, 虽然增加了算法运算时间, 但有效扩大了搜索空间, 算法的鲁棒性得到了保证.

资源交叉由设备交叉算子和工人交叉算子组成. 设备交叉算子交换两个具有相同设备基因的工序信息 ( $G_i^1$  与  $G_i^2$ ), 如图 4(a) 所示. 由于工人-工序间耦合关系基于设备建立, 设备交叉结果不违背资源能力约束. 而工人交叉算子在交换具有相同工人基因的工序信息后, 需重置违反资源能力约束的设备, 如图 4(b) 所示. 资源变异算子改变了随机基因的资源分配, 如图 4(c) 所示.

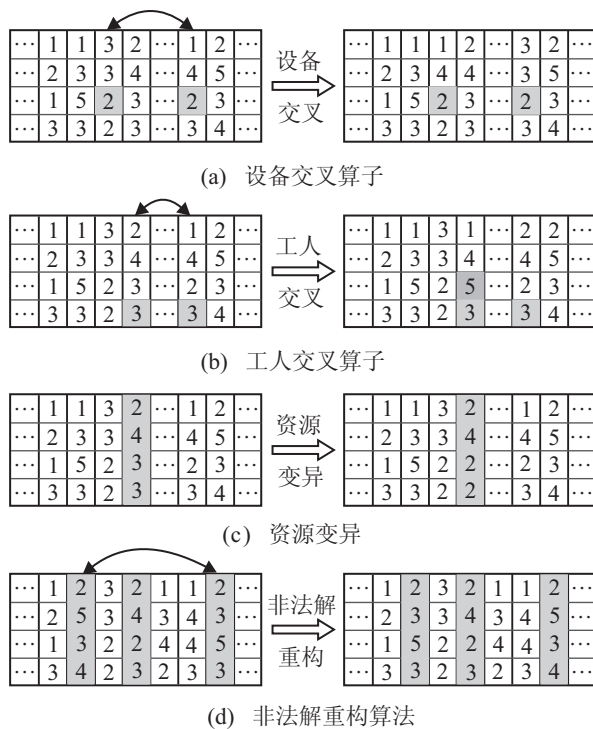


图 4 染色体进化示意图

由于 4 维编码方式导致工件、工序、资源间的强耦合关系, 上述交叉、变异操作均可能产生非法调度, 直接舍弃又将遗失其中的优秀基因片段. 对此, 本文采用一种基于 4 维编码的非法解快速重构算法, 在每次进化操作后对违反工艺约束的染色体进行快速调整: 1) 依次序遍历每个基因; 2) 对于基因  $G_i (i = 1, 2, \dots, n)$ , 当  $G_j = \min\{G_j^2 | G_j^1 = G_i^1 \wedge G_j^2 < G_i^2, j > i\} \neq \emptyset$  时,  $G_i$  与  $G_j$  交换, 如图 4(d) 所示.

作为 GA 算法的迭代运算主体结构, 进化运算复杂度直接影响算法整体运算效率. DOIGA 算法采用全概率交叉变异进化模式, 设初始种群规模为  $N$ , 第  $t$  次迭代的非劣解集规模为  $N_p$ , 分支种群规模为蚂蚁个数  $N_{ant}$ , 则  $t + 1$  次迭代进化的父辈种群规模为  $N_{evo} = N + N_{ant} + N_p$ . 每个父辈个体依次经过工

序交叉、工序变异、资源交叉、资源变异以及快速解重构, 最终得到规模为  $16N_{evo}$  的子辈种群, 运算复杂度为  $O(N_{evo} + 2N_{evo} + 4N_{evo} + 8N_{evo} + 16N_{evo}) = O(31N_{evo})$ .

### 2.4 基于 Pareto 指数的排序策略与选择算子

为充分发挥群体搜索在多目标优化方面的优势, 进化算法的选择算子多采用基于 Pareto 概念的适应值分配策略, 在 Pareto 排序基础上, 通过基于共享函数的小生境技术保留适应值共享程度较高的非劣解直至下一次进化. 但这种策略过分强调 Pareto 排序级别对存活概率的影响, 且小生境半径大小随具体问题和具体环境而变化, 尚无普适计算公式.

DOIGA 以基于个体间支配关系生成的 3 元 Pareto 指数作为染色体优劣评价标准, 将种群分为 Pareto 伴随解集  $P_{pareto}$  和进化种群  $P_{evolution}$  两部分. 一方面利用伴随解集保证已搜索到的精英个体不会在后续进化中遗失, 另一方面采用锦标赛选择算子从  $P_{evolution}$  中筛选优秀个体, 使得非支配等级较低的个体依然有存活可能, 从而增强了算法的全局搜索能力. 首先给出 Pareto 支配关系及 Pareto 指数的相关定义.

**定义 4**  $A$  支配  $B$  当且仅当

$$(f^{O_1}(A) \leq f^{O_1}(B) \wedge f^{O_2}(A) < f^{O_2}(B)) \vee (f^{O_1}(A) < f^{O_1}(B) \wedge f^{O_2}(A) \leq f^{O_2}(B)),$$

记为  $A \triangleright B$ , 其中  $f^{O_i}(A)$  表示个体  $A$  的第  $i$  个目标值.

**定义 5** Pareto 指数, 表征个体与种群中其他个体间相互支配关系的 3 维向量  $(N_g, N_e, N_b)$ . 其中:

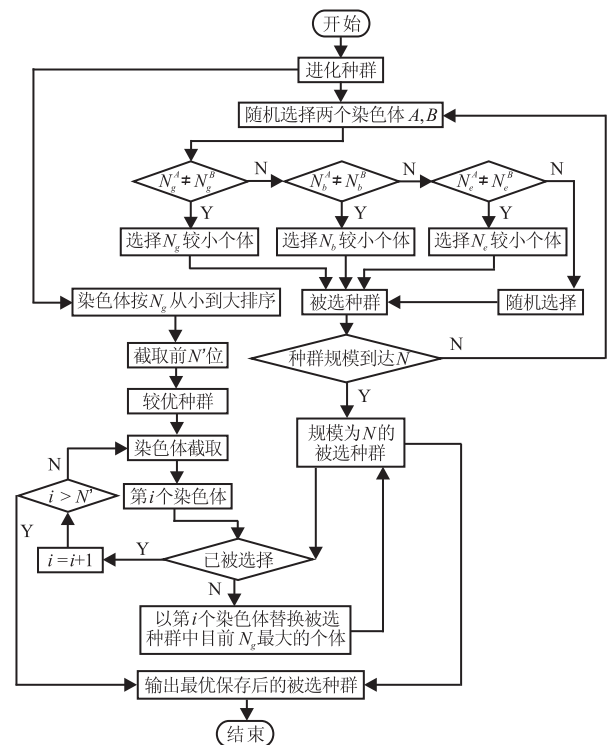


图 5 基于 Pareto 指数的锦标赛选择算子

$N_g$  为被支配因子, 该个体被种群中其他  $N_g$  个个体支配;  $N_e$  为平级因子, 该个体与其他  $N_e$  个个体互不支配;  $N_b$  为支配因子, 该个体支配  $N_b$  个个体.

种群活动化解码后, 首先对目标空间重叠个体进行唯一化处理, 隐式保证群体多样性, 避免精英个体大量繁殖而导致算法早熟. 通过计算复杂度为  $O((N^2 - N)/2)$  的个体间双目标比较, 每个染色体均得到其 Pareto 指数,  $P_{\text{pareto}}$  即为所有  $N_g = 0$  的染色体.  $N_g$  越小的个体越接近 Pareto 前沿;  $N_b$  越大, 则支配较多染色体;  $N_e$  越小, 其相对适应值分享程度越高. 采用图 5 所示的基于 Pareto 指数的锦标赛选择算子, 从所有  $N_g \neq 0$  染色体构成的  $P_{\text{evolution}}$  中选择规模为  $N$  的较优个体构成  $P'_{\text{evolution}}$ .

## 2.5 信息素更新

蚁群算法主要通过蚂蚁个体在经过路径的信息素释放量与后期信息素自然挥发量间的差值来鉴别不同路径间的差异. 只有通过信息素更新机制强化较优路径的被选择概率, 才能实现信息正反馈作用, 有效引导蚁群的搜索行为. 因此信息素变化规律是蚁群寻优实现的关键问题, 已成为众多学者的研究热点.

DOIGA 算法基于  $P'_{\text{evolution}}$  和  $P_{\text{pareto}}$ , 采用局部更新与全局更新两种方式对工艺集合进行信息素和流量的更新, 一方面模拟信息素的自然挥发, 另一方面对较优路径进行强化, 从而增强蚁群搜索过程中的信息正反馈作用. 由于成本和时间指标量纲不同, 信息素更新前先通过下式获得针对不同指标的合理信息素增量参数  $Q^{O_i}$ :

$$Q^{O_i} = 10^{\lceil \log_{10}(f^{O_i}(R_C^1)) \rceil - 2}, \quad i = 1, 2. \quad (2)$$

在避免信息素累积过快的同时保持良好的收敛速度.

1) 局部更新. 对  $P'_{\text{evolution}}$  和  $P_{\text{pareto}}$  中所有染色体所选择的工艺进行信息素、流量更新, 即

$$\begin{cases} \tau_{P_x}^{O_i} = (1 - \rho)\tau_{P_x}^{O_i} + Q^{O_i}/f^{O_i}(R_C^w), \\ n_{P_x} = n_{P_x} + 1, \end{cases}$$

$$\forall P_x \in R_C^w \cap R_C^c \in (P_{\text{pareto}} \cup P'_{\text{evolution}}).$$

2) 全局更新. 只对  $P_{\text{pareto}}$  涉及的工艺进行信息素奖励更新, 以提高精英个体经验被继承的概率, 即

$$\tau_{P_x}^{O_i} = \tau_{P_x}^{O_i} + Q^{O_i}/f^{O_i}(R_C^w),$$

$$P_x \in R_C^w \cap R_C^c \in P_{\text{pareto}}.$$

## 3 实验分析

### 3.1 非劣解集评价指标

非劣解集规模不定, 且与 Pareto 下沿的逼近程度不同, 目前尚无具有普适性的 Pareto 解集优劣评价标准. 当前经常使用的是基于目标空间的各种距离指标, 如以目标空间中 Pareto 点与原点的平均距离和,

或极值点空间距离等. 此类指标虽计算简单, 但由于  $P_{\text{pareto}}$  元素的非均衡质量, 较多劣解构成的解集反而易在平均距离指标上占优; 另外, 为兼顾收敛性与多样性, 评价指标往往多为双目标, 难以直接比较. 为便于后文实验分析, 本文提出一种基于 Pareto 指数的 Pareto 解集评价策略: 首先初始化  $M$  个  $P_{\text{pareto}}$  中所有元素的 Pareto 指数; 然后将每个解集  $P_{\text{pareto}}^i \{A_x^i\}$  的元素依次与  $P_{\text{pareto}}^j \{B_y^j\} (j > i)$  中元素进行支配关系比较, 得到各自 Pareto 指数; 最终  $N_g = 0$  的元素称为全局 Pareto 解, 每个  $P_{\text{pareto}}$  的全局 Pareto 解个数以  $R_{\text{GPareto}}$  表示, 而

$$N_g^P = \sum_{A_x^i \in P_{\text{pareto}}^i} N_g^{A_x^i}$$

称为  $P_{\text{pareto}}^i$  的被支配因子. 可见  $R_{\text{GPareto}}$  越大,  $N_g^P$  越小的  $P_{\text{pareto}}$  越逼近实际 Pareto 前沿. 当参与比较的  $P_{\text{pareto}}$  规模较大时,  $N_g^P$  的数量级远大于  $R_{\text{GPareto}}$ , 二者无法直接比较. 因此应以

$$C_{\text{pareto}} = \frac{R_{\text{GPareto}} + 1}{N_g^P / 10^{\lceil \log_{10}(\overline{N_g^P}) \rceil}}$$

作为  $P_{\text{pareto}}$  的性能评价指标, 该值越大, 该解集非劣性能越优.

### 3.2 参数实验

DOIGA 算法中对性能起关键作用的参数主要有信息素启发因子  $\alpha$ , 期望启发因子  $\beta$ , 流量启发因子  $\gamma$ , 转移概率  $q$  和信息素蒸发率  $\rho$ . 根据式 (1) 可知, 信息启发因子  $\alpha$  反映不同路径由于蚁群运动所积累的信息素浓度在指导蚂蚁路径选择决策时相对重要程度的强度; 期望启发因子  $\beta$  反映路径启发式信息在蚁群路径选择中的相对重要程度, 其大小反映了蚁群寻优过程中先验性和确定性因素的作用强度; 流量启发因子  $\gamma$  则体现蚂蚁流量对于蚂蚁路径选择时的影响程度; 转移概率  $q$  反映算法选择局部最优节点的可能; 信息素蒸发率  $\rho$  反映每次路径信息素挥发速度, 其取值对算法性能具有双重性影响. 较小的  $\rho$  使最优路径与较差路径间的信息素浓度差无法快速拉大, 从而影响算法收敛性能; 较大的  $\rho$  使落选路径信息素浓度快速衰减, 无形中压缩了蚁群的搜索空间. 因此, 上述参数的不同取值将对算法收敛速度和优化结果造成直接影响. 由于参数之间紧密耦合, 其最优取值组合的获取是一个算法性能优化的关键问题.

一般是通过对海量参数实验结果的数据统计分析得到参数的最优取值, 这样需耗费大量的实验时间. 本文通过基于均匀设计的参数实验分析不同参数组合对算法性能的影响, 以较小的实验次数确定相对最优参数组合.

均匀设计表选用  $s = 5$  的  $U_{28}(28^8)^{[16]}$ , 依据 Dorigo<sup>[17]</sup>关于 ACO 参数的实验分析,  $\alpha, \beta$  和  $\gamma$  的取值范围均限定为  $[0.5, 4]$ ,  $q$  取值范围为  $[0, 1]$ ,  $\rho$  在  $[0.1, 1]$  内取值. 各参数在取值范围内平均分布, 各取 28 个水平值, 种群规模  $N = 100$ , 分支种群规模  $N_{ant} = 20$ , 最大迭代次数  $Count = 100$ . 将 28 组参数组合分别用于独立计算文献 [18] 的 DRCJSP 双目标算例 20 次, 结果如表 1 所示. 可见当参数组合取第 17 组时平均指标最好, 且离散程度相对较低.

表 1 参数均匀实验结果

序号	$\alpha$	$\beta$	$\gamma$	$\rho$	$q$	$\bar{C}_{pareto}$
1	(1)0.50	(7)1.28	(16)2.44	(24)0.86	(25)0.89	10.44
2	(2)0.63	(14)2.19	(3)0.76	(19)0.69	(21)0.74	2.94
3	(3)0.76	(21)3.09	(19)2.83	(14)0.53	(17)0.59	3.98
4	(4)0.89	(28)4.00	(6)1.15	(9)0.36	(13)0.44	4.87
5	(5)1.02	(6)1.15	(22)3.22	(4)0.20	(9)0.30	3.99
6	(6)1.15	(13)2.06	(9)1.54	(28)0.99	(5)0.15	6.25
7	(7)1.28	(20)2.96	(25)3.61	(23)0.83	(1)0.00	8.36
8	(8)1.41	(27)3.87	(12)1.93	(18)0.66	(26)0.93	6.94
9	(9)1.54	(5)1.02	(28)4.00	(13)0.50	(22)0.78	10.90
10	(10)1.67	(12)1.93	(15)2.31	(8)0.33	(18)0.63	10.30
11	(11)1.80	(19)2.83	(2)0.63	(3)0.17	(14)0.48	6.51
12	(12)1.93	(26)3.74	(18)2.70	(27)0.96	(10)0.33	5.66
13	(13)2.06	(4)0.89	(5)1.02	(22)0.79	(6)0.19	3.62
14	(14)2.19	(11)1.80	(21)3.09	(17)0.63	(2)0.04	3.14
15	(15)2.31	(18)2.70	(8)1.41	(12)0.46	(27)0.96	10.40
16	(16)2.44	(25)3.61	(24)3.48	(7)0.30	(23)0.81	8.99
17	(17)2.57	(3)0.76	(11)1.80	(2)0.13	(19)0.67	13.51
18	(18)2.70	(10)1.67	(27)3.87	(26)0.92	(15)0.52	3.37
19	(19)2.83	(17)2.57	(14)2.19	(21)0.76	(11)0.37	5.16
20	(20)2.96	(24)3.48	(1)0.50	(16)0.59	(7)0.22	3.51
21	(21)3.09	(2)0.63	(17)2.57	(11)0.43	(3)0.07	3.63
22	(22)3.22	(9)1.54	(4)0.89	(6)0.26	(28)1.00	2.94
23	(23)3.35	(16)2.44	(20)2.96	(1)0.10	(24)0.85	4.66
24	(24)3.48	(23)3.35	(7)1.28	(25)0.89	(20)0.70	3.86
25	(25)3.61	(1)0.50	(23)3.35	(20)0.73	(16)0.56	4.52
26	(26)3.74	(8)1.41	(10)1.67	(15)0.56	(12)0.41	3.21
27	(27)3.87	(15)2.31	(26)3.74	(10)0.40	(8)0.26	6.14
28	(28)4.00	(22)3.22	(13)2.06	(5)0.23	(4)0.11	4.97

由表 2 所示方差分析可见, 不同参数取值下的算法运行结果在 0.05 的置信水平上确实存在显著差异. 因此, 通过事后比较分析 (限于篇幅, 本文未列举分析数据) 可知, 参数组 17 与参数组 1, 6, 7, 8, 9, 10, 11, 15, 16, 28 虽然在 0.05 的置信水平上不存在显著差异, 但其均值差异仍大于 0. 考虑到算法性能的最优性, 以及实验中发现上述参数组合在计算时间上并不存在明显差别, 最终选定的参数组合为组合 17, 即  $\alpha = 2.57$ ,  $\beta = 0.76$ ,  $\gamma = 1.80$ ,  $\rho = 0.13$ ,  $q = 0.67$ .

表 2 方差分析

	平方和	自由度	均方	F 值	Sig.
模型	4 652.254 <sup>a</sup>	27	172.306	1.242	0.188
组间误差	19 871.122	1	19 871.122	143.262	0.000
组内误差	73 790.924	532	138.705		
总计	98 314.300	560			

将本次实验指标最优的  $P_{pareto}$  与文献 [18] 的最优解集进行比较, 如表 3 所示. 可见, DOKIGA 算法所得  $P_{pareto}$  不仅解数量更多、分布均匀, 且绝大部分解更接近于 Pareto 下沿.

表 3 最优结果比较

DOKIGA		HGA	
$C_{max}$	Cost	$C_{max}$	Cost
52	6 886.46	52	6 886.46
53	6 865.58	53	6 865.58
54	6 821.18	54	6 865.47
55	6 740.44	55	6 740.96
56	6 695.81	56	6 688.27
57	6 450.24	57	6 450.24
59	6 426.02	—	—
60	6 409.74	60	6 409.74
61	6 328.48	61	6 380.62
62	6 325.31	—	—
63	6 310.66	63	6 368.11
64	6 157.51	64	6 272.67
65	6 154.03	65	6 206.02
66	6 051.08	66	6 147.86
67	6 028.21	67	6 028.21
68	5 991.83	—	—
69	5 987.91	—	—

### 3.3 算法性能比较

鉴于目前国内外尚无经典双资源双目标车间优化调度算例, 本文通过随机生成 10 组双资源算例对算法性能进行验证. 随机算例主要包含以下 3 类信息:

1) 工艺信息. 工件种类在  $[10, 20]$  区间内随机生成, 各工件工序数量服从均匀分布  $U[2, 10]$ , 每道工序能够在随机 2 台设备上加工, 工序的理论加工时间服从  $U[2, 30]$  分布.

2) 资源信息. 根据 ElMaraghy 等人<sup>[5]</sup>和孙志峻等人<sup>[19]</sup>的研究结论, 设备数量  $N_m$  和工人数量  $N_w$  为 10:7 时具有最佳的资源利用及调度结果, 因此设定  $N_m$  服从  $U[5, 10]$ ,  $N_w = \lceil N_m \times 0.7 \rceil$ , 每个工人所能操作的设备数服从  $U[1, 3]$  分布, 操作效率服从  $U[0.5, 1.0]$  分布, 并需要保证每台设备至少有一人能够操作.

3) 成本信息. 主要包含设备成本、工人成本及材料成本. 资源柔性越大、加工效率越高时, 使用成本越高. 参照文献 [18] 中算例的成本量纲, 设备每小时加工成本设置为  $\lceil N_F^{M^2} / \text{Mean}_{\text{Time}} \rceil$  为该设备柔性,  $\text{Mean}_{\text{Time}}$  表示该设备所有可加工工序的加工时间平均值; 工人成本设置为  $\text{sum}_e \times 10$ , 即该工人所能操作的所有设备的效率值之和  $\times 10$ ; 工件的材料成本为该工件的工序数量  $\times 10$ .

采用基于随机权值和的蚁群算法 RWACO, 以 ElMaraghy 求解 DRCJSP 的 EIGA<sup>[5]</sup> 算法为框架, 辅以 Deb 的快速 Pareto 排序策略<sup>[9]</sup> 构建的 NS-EIGA 作为对比算法, 对随机算例分别运行 20 次, 然后与算例的

表 4 算法性能比较

算例	DOIGA			RWACO			NS-EIGA		
	$\bar{R}_{GPareto}$	$\bar{N}_g^P$	$\bar{C}_{pareto}$	$\bar{R}_{GPareto}$	$\bar{N}_g^P$	$\bar{C}_{pareto}$	$\bar{R}_{GPareto}$	$\bar{N}_g^P$	$\bar{C}_{pareto}$
Pro1	1.65	1 137.15	14.37	0.00	2 598.55	0.83	0.00	2 335.50	0.62
Pro2	1.90	601.70	36.62	0.00	2 856.95	0.59	0.00	3 068.50	0.39
Pro3	2.45	1 150.55	16.85	0.00	2 530.80	0.59	0.00	2 457.75	0.59
Pro4	2.05	707.00	18.32	0.00	4 417.20	0.30	0.00	3 076.90	0.41
Pro4	1.80	2 851.55	2.60	0.75	806.35	6.66	0.20	1 360.80	1.41
Pro5	1.75	431.00	32.70	0.00	2 547.95	0.55	0.00	2 157.50	0.57
Pro6	2.50	658.25	26.72	0.00	3 476.80	0.35	0.00	3 675.35	0.32
Pro7	0.95	665.45	94.48	0.00	1 396.50	1.82	0.00	1 272.30	1.36
Pro8	1.35	431.30	18.86	0.00	2 460.25	0.58	0.00	2 625.65	0.57
Pro9	1.45	480.00	83.78	0.00	2 743.55	0.56	0.00	3 003.25	0.46
Pro10	1.65	1 137.15	14.37	0.00	2 598.55	0.83	0.00	2 335.50	0.62

60 个运行结果进行 Pareto 解集比较, 所得结果如表 4 所示. 可见, DOIGA 获得了除 Pro4 之外所有算例的全局 Pareto 解, 且平均被支配因子远小于其他两种算法, 最优的  $\bar{C}_{pareto}$  指标充分验证了 DOIGA 算法求解双目标 DRCJSP 的有效性和先进性.

#### 4 结 论

本文通过分析双资源约束作业车间调度“时间-成本”双目标优化问题的特点, 设计了继承式遗传算法, 提出一种染色体 4 维编码方式及建立在其上的基于时间窗口比较的活动化解码算法、资源交叉变异算子以及非法解快速重构等算法. 引入了蚁群算法的信息素概念, 通过生成分支种群的方式继承父辈种群的进化经验, 增强了每代遗传进化间的信息交流, 从而加速了算法的收敛. 算法中提出了基于支配关系比较的 Pareto 指数概念并构建了锦标赛选择算子, 更好地保持了进化种群的多样性. 最后通过已有算例和随机算例进行算法性能测试, 采用基于 Pareto 指数的 Pareto 解集性能评价标准, 验证了所提出算法的有效性. 关于 DOIGA 的收敛性证明等问题有待于后续研究.

#### 参考文献(References)

- [1] Nelson R T. Labor and machine limited production systems[J]. Management Science, 1967, 13(9): 648-671.
- [2] Kher H V. Examination of worker assignment and dispatching rules for managing vital customer priorities in dual resource constrained job shop environments[J]. Computers & Operations Research, 2000, 27(6): 525-537.
- [3] Salum L, Araz. Using the when/where rules in dual resource constrained systems for a hybrid push-pull control[J]. Int J of Production Research, 2009, 47(6): 1661-1677.
- [4] Bokhorst J A C, Gaalman G J C. Cross-training workers in Dual resource constrained systems with heterogeneous processing times[J]. Int J of Production Research, 2009, 47(22): 6333-6356.
- [5] Elmaraghy H, Patel V, Abdallah I B. Scheduling of manufacturing systems under dual-resource constraints using genetic algorithms[J]. J of Manufacturing Systems, 2000, 19(3): 186-201.
- [6] Schaffer J D. Multiple objective optimization with vector evaluated genetic algorithms[C]. Proc of 1st Int Conf on Genetic Algorithms and Their Application. Hillsdale: Lawrence Erlbaum Associates, 1985: 93-100.
- [7] Ishibuchi H, Murata T. A multiobjective genetic local search algorithm and its application to flowshop scheduling[J]. IEEE Trans on System, Man and Cybernetics, 1998, 28(3): 392-403.
- [8] Fonseca C M, Fleming P J. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization[C]. The Proc of 5th Int Congress on Genetic Algorithms. California: Morgan Kaufmann, 1993: 416-423.
- [9] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197.
- [10] Knowles J D, Corne D W. The Pareto archived evolutionary strategy: A new baseline algorithm for Pareto multiobjective optimization[C]. The Proc of Congress on Evolutionary Computation. Washington, 1999: 98-105.
- [11] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach[J]. IEEE Trans on Evolutionary Computation, 1999, 3(4): 257-271.
- [12] Aiex R M, Binato S, Resende M G C. Parallel GRASP with path-relinking for job shop scheduling[J]. Parallel Computing, 2003, 29(3): 393-430.
- [13] Dussutour A, Fourcassie V, Helbing D, et al. Optimal traffic organization in ants under crowded conditions[J]. Nature, 2004, 428(6978): 70-73.
- [14] Cheng R, Gen M, Tsujimura Y. A tutorial survey of job-shop scheduling problems using genetic algorithms representation[J]. Computers & Industrial Engineering, 1996, 30(4): 983-997.