

基于多目标粒子群算法的柔性作业车间调度优化方法*

王云 冯毅雄 谭建荣 李中凯

(浙江大学流体传动及控制国家重点实验室, 杭州 310027)

【摘要】 针对柔性作业车间的多目标调度问题, 构建了以制造工期、加工成本及提前/拖期惩罚值为目标函数的柔性作业车间调度模型, 提出基于密集距离排序的自适应多目标粒子群算法。采用精英策略保留进化过程中的优势个体, 基于个体密集距离降序排列进行外部种群的缩减和全局最优值的更新, 并引入小概率的变异机制以增强解的多样性和算法的全局寻优能力。最后, 将该方法应用于某机械公司的柔性作业车间多目标调度中, 仿真结果证明了该方法的有效性和适应性。

关键词: 柔性作业车间调度 多目标优化 多目标粒子群算法

中图分类号: TP278; F406.2 **文献标识码:** A **文章编号:** 1000-1298(2011)02-0190-07

Optimization Method of Flexible Job-shop Scheduling Based on Multiobjective Particle Swarm Optimization Algorithm

Wang Yun Feng Yixiong Tan Jianrong Li Zhongkai

(State Key Laboratory of Fluid Power Transmission and Control, Zhejiang University, Hangzhou 310027, China)

Abstract

To solve flexible job-shop multiobjective scheduling problem, the optimization model was set up. Considering of the makespan, manufacturing cost and earliness/tardiness penalties, a crowding distance sorting based on multiobjective particle swarm optimization algorithm was proposed. With the elitism strategy, dominant individuals were preserved in evolution process. The shrink of the external population and update of the global best were achieved by the individuals' crowding distance sorting in descending order. A small ratio mutation was introduced to enhance the diversity of solutions and the global searching capacity of the algorithm. Finally, the feasibility and validity of the method was proved by the simulation results of a flexible job-shop multiobjective scheduling in a workshop.

Key words Flexible job-shop scheduling, Multiobjective optimization, Multiobjective particle swarm optimization algorithm

引言

作业车间的调度问题一直是生产管理与组合优化等领域的热点之一。经典的作业车间调度问题(job-shop scheduling problem, 简称 JSP)的目标通常是求解一组工件的工序在一组机器上的分配。柔性作业车间调度问题(flexible job-shop scheduling problem, 简称 FJSP)是 JSP 的扩展, FJSP 与 JSP 的不同之处在于: FJSP 调度问题一般被分解成两个子

问题, 即机器选择和工序调度。由于增加了机器选择部分, FJSP 问题较 JSP 问题更加困难, 使得 FJSP 成为比 JSP 更加复杂的组合优化问题。由于柔性作业车间更加符合实际的生产环境, 因此研究 FJSP 具有重要的理论意义和实际价值。

传统 FJSP 研究主要集中在单目标调度上, 研究了多种算法, 如禁忌搜索算法^[1]、蚁群优化算法^[2]、启发式算法^[3]等。近年来, 多目标 FJSP 由于更贴近实际生产需求而引起人们的关注。目前求解柔性作

业车间多目标调度方法可分为两类: ①通过赋予每个目标不同的权重系数将多目标问题转换为单目标问题^[4-7]。②利用 Pareto 优化策略求解^[8-9]。其中很多多目标 FJSP 的处理方法是将多目标问题转换为单目标问题, 而多目标问题的特征之一是其解往往不是单个, 而是一组在多个目标之间折衷的均衡解, 即通常所说的 Pareto 最优解, 求解多目标问题的关键是找到数量足够多且分布均匀的具有代表性的 Pareto 解。

自文献 [10] 提出利用粒子群优化 (particle swarm optimization, 简称 PSO) 算法解决多目标优化问题 (multiobjective optimization problem, 简称 MOP) 以来, PSO 算法在理论研究和实际应用上都取得了大量的研究成果^[11-14]。

但是, 全局搜索能力弱和 Pareto 最优解多样性保持的复杂性依然是多目标粒子群 (multiobjective PSO, 简称 MOPSO) 存在的问题。因此, 本文采用精英策略、密集距离计算方法和小概率的变异机制, 提出基于个体密集距离排序的多目标粒子群算法 (crowding distance sorting based MOPSO, 简称 CDMOPSO), 以较小的计算复杂度获得 Pareto 前沿, 且避免早熟收敛的问题。最后用该方法求解某机械公司的 FJSP。

1 FJSP 多目标优化

1.1 FJSP 描述

FJSP 问题可以描述为: 作业车间存在 M 种工件在 N 台机器上加工, 工件 M_i 各有 J_i 道工序 (不指定工序的加工机器, 允许工序从被选机器中任意选择)。工件的工序是预先确定的, 每道工序可以在一台或多台不同的机器上加工。 R_{ijegk} 和 x_{ijk} 为决策变量。

$$R_{ijegk} =$$

$$\begin{cases} 1 & \text{(工件 } i \text{ 的第 } j \text{ 道工序和工件 } e \text{ 的第 } g \text{ 道工序} \\ & \text{在机器 } k \text{ 上执行, 若工序 } j \text{ 先于工序 } g) \\ 0 & \text{(其他)} \end{cases}$$

$$x_{ijk} = \begin{cases} 1 & \text{(工件 } i \text{ 的第 } j \text{ 道工序在机器 } k \text{ 上执行)} \\ 0 & \text{(其他)} \end{cases}$$

调度目标是为每道工序选择合适的机器, 以及确定每台机器上各工序的最佳加工顺序, 使系统的总目标达到最优。并且加工过程需要满足以下假设和约束条件: 所有机器一开始均处于空闲状态; 在零时刻, 所有的工件都可被加工; 工序一旦进行, 不能中断; 不同工件的工序之间没有先后约束, 工件之间具备相同的优先级; 各工件的准备时间和移动时间计入加工时间; 同一工件工序间的加工顺序约束; 同

一机器上一个加工任务完成后才能开始另一个任务的加工资源约束。

1.2 FJSP 多目标优化模型

本文面向的柔性作业车间多目标调度考虑 3 个优化目标 (T, C, D)。 T, C, D 分别表示制造工期、加工成本和工件提前/拖期惩罚值。

(1) 制造工期 T

$$\min T = \max P_k \quad (k = 1, 2, \dots, N) \quad (1)$$

式中 P_k ——工件在机器 k 的完工时间

式(1)表示机器 k 的完工时间取决于在其上加工的所有工件中最后一个工件的完工时间。

(2) 加工成本 C

$$\min C = \min \sum_{i=1}^M \sum_{j=1}^{J_i} \sum_{k=1}^N C_{ijk} x_{ijk} \quad (2)$$

式中 C_{ijk} ——工件 i 的第 j 道工序在机器 k 上的加工成本

(3) 提前/拖期惩罚值 D

$$\min D = \sum_{i=1}^M [r_i \max(0, D'_i - E_i) + w_i \max(0, E_i - D_i)] \quad (3)$$

通过工厂日历、订单交货期和车间加工情况分析确定工件 i 的最早交货期 D'_i 和最晚交货期时间 D_i, E_i 是工件 i 的完工时间。同时根据工件的交货优先级确定不同工件的提前惩罚系数 r_i 和拖期惩罚系数 w_i 。

(4) 约束条件

顺序约束: 工件 i 的当前工序完成后才能开始后道工序的加工。

$$E_{ijk} - E_{i(j-1)o} - m_{ijk} \geq 0$$

$$(1 < j \leq J_i, x_{ijk} = x_{i(j-1)o} = 1) \quad (4)$$

资源约束: 在机器 k 上开始一个新任务必须在一个任务完成后。

$$E_{egk} - E_{ijk} - m_{egk} \geq 0$$

$$(x_{ijk} = x_{egk} = 1, R_{ijegk} = 1) \quad (5)$$

式中 m_{ijk} ——工件 i 的第 j 道工序在机器 k 上的加工时间

E_{ijk} ——工件 i 的第 j 道工序在机器 k 上的完工时间

2 CDMOPSO 算法

文献 [12] 已经详细地归纳、总结了目前广泛应用的多种 MOPSO 算法。在 MOPSO 运算过程中, 各粒子向所经历的某个非支配的历史位置 (P_b) 学习, 同时从外部种群中按照一定规则选择一个解作为引导者 (G_b), 在外部种群中存储从开始到结束运行时粒子群所发现的非支配解。在运算过程中不断更新

外部种群,在运行结束时其中的解基本就是最终输出结果。MOPSO 算法位置更新公式仍相似于单目标优化,即

$$V_i = wV_i^{(u-1)} + c_1 \text{rand}(1) (P_b^{(u)} - X_i^{(u-1)}) + c_2 \text{rand}(2) (G_b^{(u)} - X_i^{(u-1)}) \quad (6)$$

$$X_i^{(u)} = X_i^{(u-1)} + V_i^{(u)} \quad (7)$$

式中 V ——速度 X ——粒子位置

w ——惯性因子 c_1, c_2 ——加速因子

$\text{rand}(1), \text{rand}(2)$ —— $[0, 1]$ 间的随机数

2.1 编码和解码

FJSP 问题的解包含机器选择和工序调度,因此编码也要反映这两方面的内容,为此使用基于工序和机器的两层编码方案^[15]。粒子的位置向量用 2 个相互对应的 L 维向量 $X_p[L]$ 和 $X_m[L]$ 来表示, L 是工件的总工序数。表 1 所示是一个 3×4 的 FJSP 粒子编码。

表 1 某一粒子的编码

Tab.1 Coding of a particle

粒子位置	L							
	1	2	3	4	5	6	7	8
$X_p[L]$	2	1	3	2	3	1	2	3
$X_m[L]$	4	2	1	1	3	3	3	2

解码时,先通过粒子的 $X_m[L]$ 向量值确定工序的加工机器,同时将对应的 $X_p[L]$ 向量转换为一个有序的操作表,向量中的顺序决定了工序调度的优先级。然后根据该表,将工序按其调度的优先顺序在指定的机器上以最早允许加工时间进行加工,产生调度方案。

2.2 计算位置和速度

粒子的速度矢量也由 2 部分组成,即 $V_p[L]$ 和 $V_m[L]$ 来表示,分别根据式(6)来计算。 $X_p[L]$ 和 $X_m[L]$ 根据式(7)来计算。由于 FJSP 是满足优先权约束的整数规划问题,因而粒子在更新时还需进行修正。

对计算得到的新值 $X'_p[L]$ 的各分量先按升序排列,再根据排序结果重新排列 $X_p[L]$ 的各分量。粒子的位置更新过程如表 2 和表 3 所示。

表 2 按照更新公式计算后 $X_p[L]$

Tab.2 $X_p[L]$ after updating

$X_p[L]$	2	1	3	2	3	1	2	3
$X'_p[L]$	3.1	2.2	1.4	4.5	1.8	2.5	5.2	3.8

表 3 排序更新后的新 $X_p[L]$

Tab.3 $X_p[L]$ after sorting

$X'_p[L]$	1.4	1.8	2.2	2.5	3.1	3.8	4.5	5.2
$X_p[L]$	3	3	1	1	2	3	2	2

因为 FJSP 中工序存在机器约束的情况,在 $X_m[L]$ 的粒子更新时需要进行修正:①若该分量所对应的工序只存在一台机器能够加工,则选择该工序所对应机器序号更新 $X_m[L]$ 的分量值。②若存在 l 台机器能够加工该分量对应的工序,其 $X_m[L]$ 的分量对应值为 k_1, k_2, \dots, k_l , $X_m[L]$ 的分量值 x 在计算后得到 x' , 则对 $|k_i - x'| (i = 1, 2, \dots, l)$ 进行排序,选出其最小值 k^* , 该分量值更新为 $x = k^*$ 。

2.3 多目标粒子群算法的改进

PSO 应用于多目标问题时最重要的操作是保持 Pareto 最优解集的多样性和更新粒子群的全局最优值。本文提出的 CDMOPSO 算法,借鉴密集距离计算方法对个体按密集距离排序,同时采用精英策略保留进化过程中的优势个体,从而保持 Pareto 集的多样性和更新全局最优值,并引入小概率的变异机制增强算法的全局寻优能力。主要采用如下策略:

(1) 外部种群更新和缩减

在多目标进化算法中,精英策略^[16]是在外部种群中存储每代进化过程中产生的非支配优势个体,并排除外部种群中可能产生的劣势个体,从而有利于保证多目标进化算法的收敛性和多样性。本文采用精英策略保留每代进化过程中产生的非支配个体。设内部粒子群为 P , 外部优势种群为 A , 在粒子群完成一代飞行后,首先选出 P 中的非支配个体,然后将所有非支配个体拷贝到 A , 并且删除 A 中的重复个体和被支配个体。若 A 中个体数超过其种群规模 S , 基于 Deb 在 NSGA II 中提出的个体密集距离计算方法^[17], 计算 A 中所有个体的密集距离并按降序排列。仅保留前 S 个个体,删除多余的最密集个体。否则,不对 A 进行缩减操作。该方法既通过精英策略保留运算过程中的优良粒子,同时保证了外部种群的规模,避免了运算过程中随着循环代数的增加非支配个体数增多,从而保证了运算的效率;同时,当外部种群个体数超过其规模时,删除多余的最密集个体,保留分散个体,从而有效保证了 Pareto 前沿的均匀分布。

(2) 全局最优值更新

完成外部种群 A 的更新和缩减之后,需更新内部粒子群 P 的全局最优值。 G_b 从 A 中选择一个处于最分散区域的个体,引导粒子群向 Pareto 最优前沿分散区域的进化。 G_b 的更新分为两种情况:①当 A 中仅包含边界个体时,即 A 中所有个体的密集距离为无穷大, G_b 则从中随机选择一个边界个体。②当 A 中包括若干个密集距离不为无穷大的个体时,则随机选择一个密集距离较大的个体作为

G_b , 其计算公式为

$$G_b = A_k \quad (k = \text{irnd}(n, n + \text{round}(0.1(m - n)))) \quad (8)$$

式中 m ——外部种群 A 中包含个体数量

n ——种群 A 中按密集距离降序排列的首个密集距离不为无穷大的个体序号 ($n > 1$ 且 $m \geq n$), $0.1(m - n)$ 保证选择范围在密集距离较大的个体区间内, 在该区间内随机选择一个个体作为 G_b

$\text{irnd}(n_1, n_2)$ —— $[n_1, n_2]$ 之间一个随机整数的函数

(3) 小概率的变异机制

MOPSO 算法具有收敛速度快的特点, 但过快的收敛速度会造成搜索范围受限, 导致收敛到局部最优 Pareto 前沿, 并非全局最优的 Pareto 前沿。因此本文在算法中引入 2 种基于不同编码的变异操作以摆脱局部最优点的吸引, 对粒子位置产生小范围扰动, 以提高种群多样性和扩大算法搜索空间。 c_d^t 表示 A 中第 t 代的解在第 d 维上的密集程度。

$$c_d^t = \sum_{i \in A} (x_{id}^t - \overline{x_{id}^t})^2 \quad (9)$$

式中 x_{id}^t ——第 t 代第 i 个非劣解在第 d 维上的位置

$\overline{x_{id}^t}$ ——平均位置

以 c_d^t 来反映 A 的密集程度, 当 A 变密集时, 即 $c_d^t \leq D_d$ (D_d 为事先设定的密集程度指标值), 则对内部种群 P 中所有粒子的 d 维分量 x_{id}^t 按照变异概率 p_{m_1}, p_{m_2} (p_{m_1}, p_{m_2} 为 $[0, 1]$ 区间内的较小值) 执行如下 2 种变异操作: ①对 $X_p[L]$ 进行基于工序顺序的变异, 再将变异前后适应度值较优的个体保留下来。②对 $X_m[L]$ 进行基于机器选择的变异, 把可加工该道工序的所有机器放进一个集合, 进行变异时则从该工序对应的机器集合中随机选择一个机器来替换原有机器, 再将变异前后适应度值较优的个体保留下来。

2.4 CDMOPSO 算法流程

综合上述粒子群飞行、外部种群缩减操作、全局最优值更新和小概率的变异机制, CDMOPSO 算法流程如图 1 所示。

(1) 初始化: 设置内部种群规模 N_i 与外部种群规模 M_o , 设置最大迭代次数 G_{\max} 。

(2) 令迭代次数 $G = 0$, 初始化内、外种群: ①随机生成初始的内部粒子群 P , 其中的变量在允许范围内随机取值, 设置粒子速度初值为 0, 局部最优值等于变量值。②根据 P 中变量值计算目标函数值。

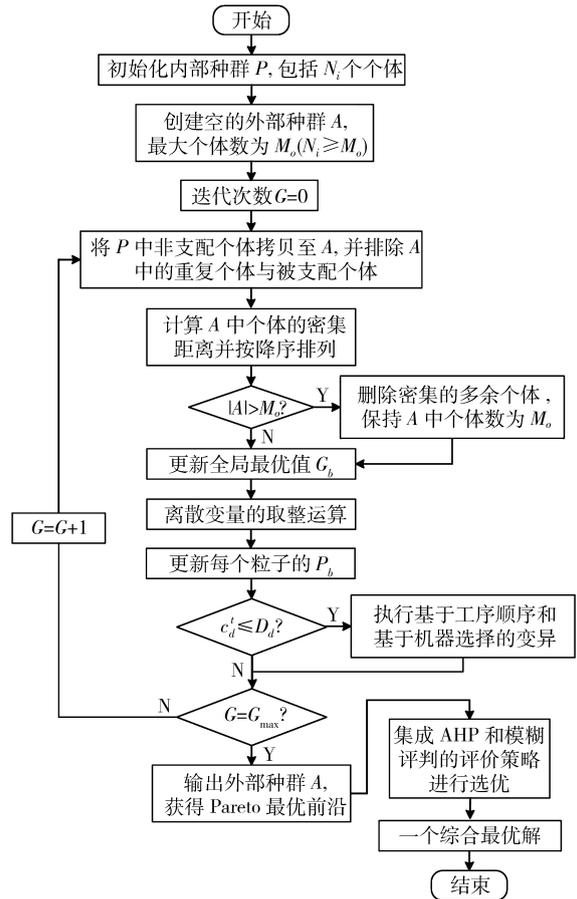


图 1 基于 CDMOPSO 算法流程

Fig. 1 Flow chart of CDMOPSO

③建立一个空的外部种群 A 。

(3) 更新外部种群 A , 若 A 中个体数超过其最大允许数目 M_o , 采用基于个体密集距离降序排列的方法缩减外部种群。

(4) 根据全局最优值更新策略更新 G_b ; 根据式(6)、(7), 更新 P 中的速度和位置, 并对粒子新位置和局部最优位置进行, 从而更新粒子的 P_b 。

(5) 判断外部种群 A 是否密集 ($c_d^t \leq D_d$)。若否, 执行步骤(6); 若是, 则分别按照 p_{m_1}, p_{m_2} 对内部粒子群 P 执行基于工序顺序和基于机器选择的变异。

(6) 判断是否达到最大循环代数 ($G \geq G_{\max}$)。若否, 令 $G = G + 1$, 返回步骤(3)继续运行; 若是, 输出外部种群, 获得 Pareto 最优解集。

(7) 采用层次分析法和模糊评判集成的策略^[18]进行选优, 得出一个综合最优解。

3 实例应用与分析

该方法已在某机械公司的 ERP 系统中生产管理和柔性作业车间调度模块得到实际应用。系统采用 Powerbuilder 9.0、C++ 为开发平台和 SQL SERVER 2000 为后台数据库, 采用 Client/Server 架

构。以该公司的磨具车间为例,进行柔性作业车间调度多目标优化。该系统通过工件工艺卡管理获取工件加工的基础参数,如图2所示。对该车间作业计划的原始数据进行计算处理后得到表4和表5。表4是调度问题的原始数据,包括工件的每道工序对应加工机器、加工时间(min)、加工成本(元);表5是工件交货期和提前/拖期惩罚系数表。

以制造工期最短、加工成本最低及提前/拖期惩罚最少为目标函数,进行FJSP的多目标优化。设置内部种群 $N_i = 100$ 、外部种群 $M_o = 20$ 、循环代数 $G = 800$ 、惯性因子 $w = 0.3$ 、加速因子 $c_1 = c_2 = 2$ 、变异概率 $p_{m1} = p_{m2} = 0.125$,图3为利用获得的 Pareto 解的 Matlab 7.0 仿真图。



图2 工件工艺卡管理界面

Fig.2 Management of process card

表4 工序的加工机器、时间、成本

Tab.4 Machining equipment, time and fee of jobs

工件	工序					
	1	2	3	4	5	6
1	1,4,7.2	3,12,19.8	4,24,43.2	2,28,42.0	6,12,19.8	5,24,28.8
	3,44,46.2					
2	4,32,55.2	1,20,36.0	5,40,81.0	6,40,32.0	3,40,66.0	2,16,24.0
	3	2,16,27.0				
4		4,20,35.0	3,20,39.0	1,20,42.0	3,36,64.8	3,26,38.0
	5	1,36,59.4	4,12,26.1		6,48,72.3	
6		6,40,75.0	1,20,46.4	5,20,45.0	6,16,27.2	4,28,50.4
	6	4,12,21.6	2,12,23.4	6,36,64.8	3,40,54.0	5,16,34.8
		6,48,57.6	1,44,48.4			

表5 工件交货期和提前/拖期惩罚系数

Tab.5 Due date and earliness/tardiness penalty of jobs

工件	最早交货期/min	最晚交货期/min	提前惩罚系数	拖期惩罚系数
1	210	230	0.2	0.4
2		230		0.2
3		155		0.5
4	210	225	0.2	0.2
5		230		0.3
6		215		0.2

采用层次分析法和模糊评判集成的策略,对得到的 Pareto 集进行选优,通过计算可得到该柔性车间多目标调度优化问题的综合最优方案。图4为系统所选出的综合最优方案对应的甘特图,图中的数

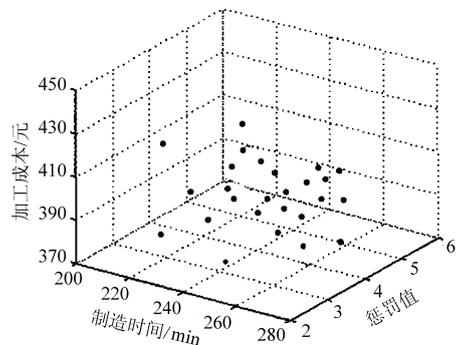


图3 FJSP 多目标优化的 Pareto 解

Fig.3 Pareto solutions of FJSP MOPSO

字表示工件工序号。

对于多目标优化算法,通常可以用分布度 Δ ^[17] 来表示所求解集的分布性,以变化率 Δ_t 来表示算法运行的稳定性。个体分布越均匀, Δ 越接近零。 Δ_t

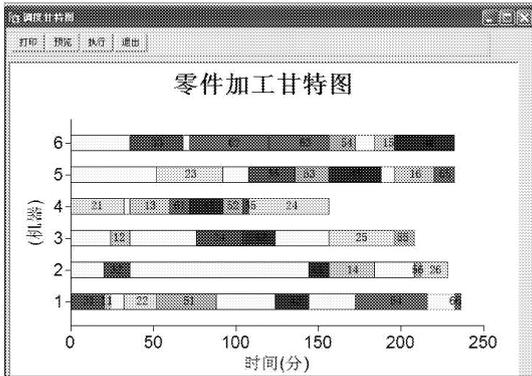


图 4 Pareto 综合最优解对应的甘特图

Fig.4 Gantt chart of optimal scheduling

值越小表明算法运行稳定性越好。为了验证该算法在解决柔性车间多目标调度优化问题上的优越性,与 VEGA 和 NSGA II 算法进行了计算比较。同时运行上述 3 种算法后,将它们找到的 Pareto 解拷贝到一个集合中,删除集合中的被支配个体,然后计算 3 种算法保留的 Pareto 解的数量在此集合中所占的比例。从所占比例和分布情况来考察算法在求解多目标 FJSP 上的有效性。表 6 为应用本文 CDMOPSO 算法、VEGA 和 NSGA II 算法各运行 20 次平均值的对比结果,可以看出 CDMOPSO 不仅 Pareto 解所占比例要多于 VEGA 和 NSGA II 算法,而且具有较好的分布性和稳定性。

采用层次分析法和模糊评判集成的策略,对 3 种算法得到的 Pareto 解进行优选,可得到各算法的综合最优解。表 7 为 3 种算法运行 20 次得到的综合最优解对比结果。计算结果表明,CDMOPSO 与 VEGA、NSGA II 算法相比,综合最优解的性能最

表 6 Pareto 所占比例和分布情况对比

Tab.6 Comparison of Pareto solutions ratio and distribution

算法	Pareto 解所占比例/%	分布度 Δ	变化率 Δ_r
CDMOPSO	43.28	0.5214	0.0279
VEGA	25.37	0.6215	0.0346
NSGAI	31.35	0.5814	0.0312

表 7 综合最优解性能比较

Tab.7 Comparison of optimal solutions

优化目标	CDMOPSO		VEGA		NSGA II	
	最优	平均	最优	平均	最优	平均
加工成本/元	392.40	410.15	401.30	425.75	397.50	422.25
制造工期/min	227.00	238.30	235.00	248.30	231.00	244.90
惩罚值	3.70	4.02	4.30	4.67	4.20	4.43

好,能更好地解决柔性作业车间多目标调度问题。

4 结束语

本文在构建柔性车间多目标调度模型的基础上,结合精英策略和个体密集距离计算,改进 MOPSO 算法的外部种群缩减和全局最优值更新,并引入小概率的变异机制来增强算法的全局寻优能力,提出基于个体密集距离排序的多目标粒子群算法,以制造工期、加工成本和提前/拖期惩罚值为目标进行优化,得到 FJSP 的综合最优解。最后以某机械公司的车间调度为例,与 VEGA 和 NSGA II 算法进行比较,结果表明提出的方法能有效地解决柔性作业车间调度的多目标优化问题。

参 考 文 献

- Brandimarte P. Routing and scheduling in a flexible job shop by tabu search [J]. Annals of Operations Research, 1993, 22(2):157 ~ 183.
- 张维存,郑丕谔,吴晓丹. 蚁群遗传算法求解能力约束的柔性作业车间调度问题[J]. 计算机集成制造系统,2007, 13(2):333 ~ 337.
Zhang Weicun, Zheng Pi'e, Wu Xiaodan. Solution to flexible job shop scheduling problems with capacitated constraints based on ant colony & genetic algorithms[J]. CIMS, 2007,13(2): 333 ~ 337. (in Chinese)
- Mati Y, Rezg N, Xie X L. An integrated greedy heuristic for a flexible job shop scheduling problem[C] // Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, 2001,4:2 534 ~ 2 539.
- Xing Lining, Chen Yingwu, Yang Kewei. Multi-objective flexible job shop schedule: design and evaluation by simulation modeling[J]. Applied Soft Computing, 2009,9(1): 362 ~ 376.
- 余建军,孙树栋,郝京辉. 免疫算法求解多目标柔性作业车间调度研究[J]. 计算机集成制造系统, 2006, 12(10): 1 643 ~ 1 651.
Yu Jianjun,Su Shudong,Hao Jinghui. Multi objective flexible job-shop scheduling based on immune algorithm[J]. CIMS, 2006, 12(10): 1 643 ~ 1 651. (in Chinese)
- Zhang Guohui, Shao Xinyu, Li Peigen, et al. An effective hybrid particle swarm optimization algorithm for multi-objective flexible job-shop scheduling problem[J]. Computers & Industrial Engineering, 2009,56(2): 1 309 ~ 1 318.
- Xia Weijun, Wu Zhiming. An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems

- [J]. Computers & Industrial Engineering, 2005, 48(2): 409 ~ 425.
- 8 Kacem I, Hammadi S, Borne P. Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic[J]. Mathematics and Computers in Simulation, 2002, 60(3 ~ 5): 245 ~ 276.
- 9 魏巍, 谭建荣, 冯毅雄, 等. 柔性工作车间调度问题的多目标优化方法研究[J]. 计算机集成制造系统, 2009, 15(8): 1 592 ~ 1 598.
- Wei Wei, Tan Jianrong, Feng Yixiong, et al. Multi-objective optimization method research on flexible job shop scheduling problem[J]. CIMS, 2009, 15(8): 1 592 ~ 1 598. (in Chinese)
- 10 Coello C, Salazar M. MOPSO: a proposal for multiple objective particle swarm optimization [C] // Proceedings of 2002 Congress on Evolutionary Computation (CEC), 2002, 2: 1 051 ~ 1 056.
- 11 Li X. A non-dominated sorting particle swarm optimizer for multiobjective optimization [C] // Lecture Notes in Computer Science. Berlin: Springer-Verlag, 2003, 31(2): 37 ~ 48.
- 12 Durillo J J, Garcia-Nieto J, Nebro A J, et al. Multi-objective particle swarm: optimizers an experimental comparison [C] // Proceedings 5th International Conference, EMO 2009, Lecture Notes in Computer Science, Berlin: Springer-Verlag, 2009, 5467: 495 ~ 509.
- 13 Abido M A. Multiobjective particle swarm optimization for environmental/economic dispatch problem [J]. Electric Power System Research, 2009, 79(7): 1 105 ~ 1 113.
- 14 徐杰, 黄德先. 基于混合粒子群算法的多目标车辆路径研究[J]. 计算机集成制造系统, 2007, 13(3): 573 ~ 579.
- Xu Jie, Huang Dexian. Hybrid particle swarm optimization for vehicle routing problem with multiple objectives [J]. Computer Integrated Manufacturing System, 2007, 13(3): 573 ~ 579. (in Chinese)
- 15 贾兆红, 陈华平, 唐俊, 等. 面向多目标的自适应动态概率粒子群优化算法[J]. 系统仿真学报, 2008, 20(18): 4 959 ~ 4 963.
- Jia Zhaohong, Chen Huaping, Tang Jun, et al. Self-adaptive dynamic probabilistic particle swarm optimization algorithm for multiple objectives [J]. Journal of System Simulation, 2008, 20(18): 4 959 ~ 4 963. (in Chinese)
- 16 郑向伟, 刘弘. 多目标进化算法研究进展[J]. 计算机科学, 2007, 34(7): 187 ~ 192.
- Zheng Xiangwei, Liu Hong. Progress of research on multi-objective evolutionary algorithms [J]. Computer Science, 2007, 34(7): 187 ~ 192. (in Chinese)
- 17 Deb K, Pratap A, Agarwal S. A fast and elitist multi-objective genetic algorithm: NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182 ~ 197.
- 18 吴秀丽, 孙树栋, 余建军, 等. 多目标柔性作业车间调度决策精选机制研究[J]. 中国机械工程, 2007, 18(2): 161 ~ 165.
- Wu Xiuli, Sun Shudong, Yu Jianjun, et al. A multi-objective scheduling decision making model for the flexible job shop [J]. China Mechanical Engineering, 2007, 18(2): 161 ~ 165. (in Chinese)
- 19 王雷, 唐敦兵, 万敏, 等. 激素调节机制 IAGA 在作业车间调度中的应用[J]. 农业机械学报, 2009, 40(10): 199 ~ 202.
- Wang Lei, Tang Dunbing, Wan Min, et al. Job-shop scheduling problem based on improved adaptive genetic algorithm with hormone modulation mechanism [J]. Transactions of the Chinese Society for Agricultural Machinery, 2009, 40(10): 199 ~ 202. (in Chinese)
- 20 刘丽琴, 张学良, 谢黎明, 等. 基于动态聚集距离的多目标粒子群优化算法及其应用[J]. 农业机械学报, 2010, 41(3): 189 ~ 194.
- Liu Liqin, Zhang Xueliang, Xie Liming, et al. Multi-objective particle swarm optimization algorithm based on dynamic crowding distance and its application [J]. Transactions of the Chinese Society for Agricultural Machinery, 2010, 41(3): 189 ~ 194. (in Chinese)