

改进狼群算法求解多目标柔性作业车间调度问题*

陈嘉朋, 张宏立[†], 王聪, 马萍

(新疆大学 电气工程学院, 新疆 乌鲁木齐 830017)

摘要: 针对传统智能优化算法求解多目标柔性作业车间调度时存在算法后期收敛速度慢、易陷入局部最优的问题, 本文提出一种将量子粒子群算法中的三大重要性能参数和狼群算法融合的混合优化算法. 首先, 构建以最大完工时间、机器总负荷和瓶颈机器负荷为优化目标的多目标数学模型; 其次, 采用高斯分布的概率密度函数产生随机变量进行种群初始化操作, 以提高初始种群的多样性和质量; 利用邻域结构搜索策略不断调整最佳序列, 算法的全局搜索性能得以提高; 最后, 通过物元分析法对种群进行更新, 提高种群的自适应能力. 通过与多种智能优化算法的仿真实验对比可知, 本文所提出的混合狼群算法对求解多目标柔性作业车间调度问题具有可行性和优势.

关键词: 狼群算法; 量子粒子群算法; 多目标柔性作业; 高斯分布; 邻域搜索; 自适应机制

DOI: 10.13568/j.cnki.651094.651316.2020.11.19.0003

中图分类号: TP301.6 **文献标识码:** A **文章编号:** 2096-7675(2022)01-0042-07

引文格式: 陈嘉朋, 张宏立, 王聪, 马萍. 改进狼群算法求解多目标柔性作业车间调度问题[J]. 新疆大学学报(自然科学版)(中英文), 2022, 39(1): 42-48+73.

英文引文格式: CHEN Jiapeng, ZHANG Hongli, WANG Cong, MA Ping. Improved wolf pack algorithm for solving multi-objective flexible job shop scheduling problem[J]. Journal of Xinjiang University(Natural Science Edition in Chinese and English), 2022, 39(1): 42-48+73.

Improved Wolf Pack Algorithm for Solving Multi-Objective Flexible Job Shop Scheduling Problem

CHEN Jiapeng, ZHANG Hongli, WANG Cong, MA Ping

(School of Electrical Engineering, Xinjiang University, Urumqi Xinjiang 830017, China)

Abstract: In order to solve the problem of slow convergence rate and easy to get into local optimum when solving multi-objective flexible job shop scheduling problem, with traditional intelligence optimization algorithm, this paper proposes a hybrid optimization algorithm that fuses three important performance parameters of QPSO with the wolf pack algorithm. Firstly, a multi-objective mathematical model with maximum completion time, total machine load and bottleneck machine load as optimization indexes is constructed. Secondly, the probability density function of Gaussian distribution is used to generate random variables for population initialization so as to improve the diversity and quality of the initial population. Neighborhood structure search strategy is used to adjust the optimal sequence, and the global search performance of the algorithm is improved. Finally, the matter-element analysis method is used to update the population and to improve the adaptive ability of the population. By comparing with the simulation experiment of many intelligent optimization algorithms, we can see that the hybrid wolf pack algorithm proposed in this paper is feasible and advantageous for solving the multi-objective flexible job shop scheduling problem.

Key words: wolf pack algorithm; quantum particle swarm optimization; multi-objective flexible operation; Gaussian distribution; neighborhood search; adaptive mechanism

0 引言

柔性作业车间调度问题(Flexible Job-shop Scheduling Problem, FJSP)^[1]是一类复杂的NP-hard优化问题, 其难点在于需要同时考虑作业车间的加工次序和加工机器. 随着对FJSP的研究不断深入, 更符合实际需要的

* 收稿日期: 2020-11-19

基金项目: 国家自然科学基金(51767022; 51967019); 新疆维吾尔自治区自然科学基金项目(2019D01C082).

作者简介: 陈嘉朋(1996-), 男, 硕士生, 从事车间调度、优化算法的研究, E-mail: 597403586@qq.com.

[†] 通讯作者: 张宏立(1972-), 男, 博士, 教授, 从事复杂生产过程、非线性系统的混沌动力学分析与调度的研究, E-mail: zh-lxju@163.com.

多目标柔性作业车间调度问题 (Multi-objective Flexible Job-shop Scheduling Problem, MOFJSP) [2] 成为目前的研究热点. 在研究多目标柔性作业车间调度问题时, MOFJSP 满足更多的约束条件, 同时各个目标函数之间互相制约, 导致各个目标函数值很难同时达到最优. 所以, MOFJSP 是更难的 NP-hard 优化问题.

许多学者针对 MOFJSP 进行了广泛和深入的研究. Huang [3] 等人将粒子群算法和遗传算法巧妙融合, 通过教与学的方式体现混合算法求解 MOFJSP 的有效性; 朱光宇 [4] 等人将灰熵并行分析引入粒子群算法对多目标柔性作业车间调度问题进行求解, 通过测试验证了灰熵并行法的有效性; 朱伟 [5] 等人运用遗传算法 (Genetic Algorithm, GA) 来优化以加工总成本和最大加工时间为目标的 MOFJSP, 通过对算法算子、编码结构、自适应变异规则的改进和更新, 算法和模型的鲁棒性和有效性得以提高; 赵博选 [6] 等人提出一种多策略融合 Pareto 人工蜂群算法来求解 MOFJSP, 多种搜索策略互相融合, 不仅使算法实现了人工蜂群的自主与协同搜索, 而且达到了全局探索与局部寻优的平衡效果. 上述智能优化算法为 MOFJSP 的求解开辟了新的研究方向和思路, 但算法求解全局最优解的能力有待进一步提高.

为解决传统群智能优化算法在求解 MOFJSP 时存在易陷入局部最优、算法后期收敛速度慢等问题, 本文使用吴虎胜等人提出的一种新型群智能优化算法——狼群算法 [7], 并将量子粒子群算法中的三个重要参数与其融合, 得到一种新的混合优化算法对 MOFJSP 进行求解. 狼群算法模拟狼群猎食行为及猎物分配方式, 抽象出“游走”“召唤”“围攻”3 种智能行为和“胜者为王”的头狼产生规则及“强者生存”的狼群更新机制. 狼群算法结构清晰, 寻优策略丰富, 特别是针对多峰、高维的目标函数有着较好的求解性能, 这些优势使算法在时间序列预测 [8]、故障诊断 [9] 等方面有着广泛的应用, 但狼群算法在寻优过程中, 也存在算法后期收敛速度慢、易陷入局部最优的问题.

本文针对狼群算法寻优过程中存在的问题, 提出一种将量子粒子群算法中能够引导粒子以较快的收敛速度, 在一定的搜索范围内朝着全局最佳位置不断靠近的三种性能参数与狼群算法融合的混合优化算法来求解 MOFJSP. 创新点主要为:

(1) 种群初始化. 将高斯分布的概率密度函数运用于种群初始化, 以得到质量较高的初始种群.

(2) 搜索策略. 将量子粒子群算法中的吸引子、收缩扩张系数、势阱特征长度三个重要参数和狼群算法融合, 增强算法的全局搜索性能和收敛速度; 采用物元分析法更新种群, 提高种群的自适应能力.

(3) 最优解操作. 通过局部邻域搜索, 给予最佳序列充分的计算资源, 以增强算法的全局搜索性能.

1 MOFJSP 问题描述和数学模型建立

1.1 问题描述

FJSP 描述如下: m 个工件在 n 台机器上加工, 每个工件至少有一道加工工序, 每道工序可以选择不同的机器进行加工, 不同工序的加工时间取决于所选机器的性能. 表 1 为 FJSP 的一个加工实例.

表 1 FJSP 加工实例

工件	工序	可选机器			
		M_1	M_2	M_3	M_4
J ₁	O ₁₁	2	6	7	2
	O ₁₂	3	-	-	4
	O ₁₃	2	-	1	2
J ₂	O ₂₁	5	3	-	3
	O ₂₂	3	-	4	2
	O ₂₃	2	4	1	-

注: 1. “-”表示此工序不能在此机器进行加工. J 代表工件, M 代表加工机器, O_{ij} 代表工件 i 的第 j 道工序, 表中的数字为不同工序在不同机器上的加工时间. 2. 此表为部分柔性作业车间调度问题实例, 含义是某些工序只能被部分机器加工.

1.2 建立 FJSP 数学模型

1.2.1 数学变量定义

m 为工件数; n 为机器数; i 为工件序号, $i = 1, 2, 3, \dots, m$; j 为机器序号, $j = 1, 2, 3, \dots, n$; C_i 为每个工件的完工时间; C_{\max} 为最大完工时间; O_{ih} 为第 i 个工件的第 h 道工序; M_{ih} 为第 i 个工件的第 h 道工序的可选加工机器

数; M_{jih} 为第 i 个工件的第 h 道工序在机器 j 上加工; P_{jih} 为第 i 个工件的第 h 道工序在机器 j 上的加工时间; T_{ih} 为第 i 个工件的第 h 道工序开始加工时间; C_{ih} 为第 i 个工件的第 h 道工序加工完成时间; S_{jih} 为 M_{jih} 对应的加工时间; L 为工序序号, $L=1,2,3,\dots,h_i$; $x_{jih}=1$, 工序 O_{ih} 选择机器 j 进行加工; $x_{jih}=0$, 工序 O_{ih} 不选择机器 j 进行加工; $y_{ihjkl}=1$, 工序 O_{ih} 先于工序 O_{jkl} 进行加工; $y_{ihjkl}=0$, 工序 O_{ih} 晚于工序 O_{jkl} 进行加工.

1.2.2 MOFJSP调度指标

(1) 最大完工时间

最大完工时间是指所有工件均被加工之后, 最后离开机器的一道工序所对应的完工时间. 性能指标通常表述为:

$$C_m = \min \left(\max_{1 \leq i \leq m} (C_i) \right) \quad (1)$$

其中: C_i 代表每个工件的完工时间.

(2) 机器总负荷

机器总负荷是指所有工件均被加工完成之后, 工件在机器上的加工时间总和.

$$w_t = \min \left(\sum_{j=1}^n \sum_{i=1}^m p_{jih} x_{jih} \right) \quad (2)$$

(3) 瓶颈机器负荷

瓶颈机器负荷是指所有工序被加工完成之后, 负荷(加工时间)最多的机器所用的时间.

$$w_m = \min \left(\max_{1 \leq j \leq n} \sum_{i=1}^m p_{jih} x_{jih} \right) \quad (3)$$

2 算法介绍

狼群算法(Wolf Pack Algorithm, WPA)是基于狼群智能行为的新型优化算法, 算法基本思想是“任务分配, 各尽其职”和“弱肉强食, 适者生存”. 算法意在通过模拟自然界的生物种群行为由概率模型搜索最优解. 量子粒子群算法(Quantum Particle Swarm Optimization, QPSO)是一种考虑量子行为的粒子群算法, 该算法相较于传统的粒子群算法, 具有更好的收敛速度和全局搜索性能. 将QPSO中的三种重要性能参数与WPA相结合, 以形成改进狼群算法对多目标柔性作业车间调度问题进行求解. 狼群位置更新分为三个阶段, 分别为探狼游走、头狼召唤和猛狼围攻. 三个阶段的主要改进位置更新公式如(4)、(5)和(6)所示:

$$x_{jd}^p = x_{jd} + \beta \sin \left(2\pi \times \frac{p}{h} + w \right) \times \text{step}_a^d \quad (4)$$

式(4)为探狼游走位置更新公式. 探狼在搜索空间中朝着猎物气味浓度更深的方向不断游走, 以获取最佳位置. 当探狼达到最大游走次数时, 选择最佳目标函数值的狼为新头狼. 为使探狼进行更为精细的搜索, 扩大种群的搜索范围, 避免算法陷入局部最优, 将收缩扩张系数 β 和归一化因子 w 引入探狼游走行为.

$$x_{id}^{\text{new}} = x_{id} + s_{(i)} \times \text{step}_b^d \times ((g_d - x_d) / |g_d - x_d|) \quad (5)$$

式(5)为猛狼奔袭更新公式. QPSO中粒子 i 的吸引子 $s_{(i)}$ 能够使粒子以较快的速度朝着最佳粒子不断移动, 以提高算法的收敛速度和全局搜索性能.

$$x_{id}^* = x_{id} + L_i \text{step}_c^d \times |g_d - x_{id}| \quad (6)$$

狼群在围攻猎物的过程中, 容易超出设定的搜索空间范围, 得到不可行解, 从而导致算法的求解性能下降. 将势阱特征长度 L_i 引入狼群围攻行为的位置更新公式, 如式(6). 势阱特征长度 L_i 能够使参与围攻行为的个体处在有效的搜索空间内.

式(4)~式(6)中, step_a^d 为游走步长, step_b^d 为奔袭步长, step_c^d 为攻击步长; w 为偏移因子; h 为探狼 j 每一次游走方向数, p 为探狼 j 每一次游走的第 p 个方向; g_d 为当前头狼第 d 维的位置; β 为QPSO中的收缩扩张系数, $s_{(i)}$ 为吸引子, L_i 为势阱特征长度. 在WPA中加入三个性能参数使得寻优个体在一定的搜索空间内, 以较快收敛速度朝着全局最佳个体不断靠近.

2.1 种群更新机制

采用物元分析法, 对猎食之后的种群进行更新和保存操作, 以满足更优的解进入种群下一次迭代. 在物元分析中, 物元 $R=(M,C,X)$, M 代表物元的名称, C 代表特征, X 代表特征对应的物元量值. 将个体对应的位置放入物元特征, 将位置更新操作放入物元名称, 所得到的新适应度函数值作为物元量值. 矩阵 R_{mn} 为 n 个 m 维物元矩阵.

$$R_{mn} = \begin{bmatrix} M_1 & M_2 & \dots & M_n \\ C_1 & fit(C_1^1) & fit(C_1^2) & \dots & fit(C_1^n) \\ C_2 & fit(C_2^1) & fit(C_2^2) & \dots & fit(C_2^n) \\ \dots & \dots & \dots & \dots & \dots \\ C_m & fit(C_m^1) & fit(C_m^2) & \dots & fit(C_m^n) \end{bmatrix}$$

其中: $C_1, C_2, C_3 \dots C_m$ 为捕猎之后个体对应的位置; $M_1, M_2, M_3 \dots M_n$ 为不同位置下的不同更新策略; $fit(C_m^n)$ 为不同位置更新之后的适应度函数值.

以 C_1 为例:

$$R_{1n} = \begin{bmatrix} M_1 & M_2 & \dots & M_n \\ C_1 & fit(C_1^1) & fit(C_1^2) & \dots & fit(C_1^n) \end{bmatrix}$$

其中: $C_1 = (x_1^1, x_1^2 \dots x_1^d)$, d 代表维数; $M_1 \sim M_n$ 表示在 C_1 的基础上, 每一维增加一个 $(-1,1)$ 上的随机数; 得到 $C_1^1 = C_1 + \text{rand}(-1,1)$, $C_1^2 = C_1 + \text{rand}(-1,1) \dots$ 以此类推. 从 $fit(C_1^1) \sim fit(C_1^n)$ 选择最大适应度函数值且个体满足边界限制的位置作为个体新位置. 通过物元分析法, 丰富了种群的多样性和质量.

3 编码和解码

3.1 种群初始化

初始种群的质量对种群迭代过程中的收敛速度以及全局最优解的求解起到重要的作用, 生成一个较高质量的初始种群十分关键. 本文采用高斯分布的概率密度函数^[10]产生随机变量来调整加工序列, 从而对种群进行初始化. 采用的概率密度函数为:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right) \tag{7}$$

其中: μ 为均值, 表示概率密度函数分布曲线的对称轴; σ 为标准差, 表示对称轴到左右两边曲线的某一宽度.

初始化操作

Step1 设置 $x_1 = \text{rand}(0,1)$, $\sigma_i = \text{rand}(0,1)$, $\mu_i = \text{rand}(-1,1)$, $i = 1, 2, 3 \dots L$, L 为总工序数. 保持 x_1 不变, 将 L 组 (σ_i, μ_i) 带入 $f(x)$, 得到一组随机序列 $[f_1(x_1), f_2(x_1) \dots f_L(x_1)]$, 对应初始加工工序;

Step2 将 $[f_1(x_1), f_2(x_1) \dots f_L(x_1)]$ 进行升序排列, 同时将初始加工工序随着变量的升序进行调整, 得到新的加工工序.

通过重复Step1到Step2, 直到产生与种群规模相同的初始加工工序. 利用高斯分布的概率密度函数产生随机数序列的方式来进行种群初始化, 可以使初始种群个体离散地分布在解空间内, 有效提高了初始种群的质量和多样性. 种群初始化过程如表2所示.

表 2 种群初始化

位置	1	2	3	4	5	6
初始工序	O ₁₁	O ₁₂	O ₂₁	O ₂₂	O ₃₁	O ₃₂
随机序列	0.2	0.1	0.3	0.45	0.15	0.4
序列升序	0.1	0.15	0.2	0.3	0.4	0.45
新工序1	O ₁₂	O ₃₁	O ₁₁	O ₂₁	O ₃₂	O ₂₂

3.2 编码和解码

为更好满足工艺路线的约束, 本文采用双层编码方式进行编码. 基于表1, 工序编码为(113322), 表示O₁₁、O₁₂、O₃₁、O₃₂..., 以此类推; 设机器编码为(113212), 表示O₁₁选择机器1, O₁₂选择机器1, O₁₃选择机器3, O₂₁选择机器2...以此类推 (O_{ij}代表工件*i*的第*j*道工序).

工序解码

根据Logistic映射方程^[11],产生一组随机均匀的随机数 S_1 . S_1 对应相应的加工工序. 将 S_1 升序,并得到对应索引值. 升序时,若 S_1 出现相同的数值,以左(小数值)进行优先排序. S_1 进行升序后,计算 $S(\text{new}) = S_1 \times (G \div V)$, G 为工件总数, V 为工序总数. 当 $S(\text{new})$ 为小数时,将其朝着比自身大的方向进行取整. $S(\text{new})$ 对应相应的工件号. 表3为工序解码过程.

表 3 工序解码

S_1	0.84	0.96	0.8	0.6	0.72	0.98
工件号	J_1	J_1	J_2	J_2	J_3	J_3
S_1 升序	4	5	3	1	2	6
$S(\text{new})$	2	3	2	1	1	3
工件号	J_2	J_3	J_2	J_1	J_1	J_3
工序	O_{21}	O_{31}	O_{22}	O_{11}	O_{12}	O_{32}

机器解码

设工件 i 的第 h 道工序为 u ;第 i 个工件的第 h 道工序开始加工时间为 $ST(u)$,加工时间为 T_p ,则完工时间为 $ST(u) + T_p$;该工序的工件前任工序和机器前任工序分别用 $JP(u)$ 和 $MP(u)$ 表示. 则有:

$$ST_{(u)} = \max(ST_{JP(u)} + T_{pJP(u)}, ST_{MP(u)} + T_{pMP(u)}) \quad (8)$$

本文采用贪婪策略对机器进行解码. 选择使 $ST(u) + T_p$ 最小的机器加工,若出现机器的加工时间相同,则选择使 T_p 最小的机器加工.

3.3 最佳序列更新机制

本文采用邻域结构搜索策略,给予最佳序列充分的计算资源,不断提高算法寻求全局最优解的能力和收敛速度.

NS1: 随机选择工序码的两个位置进行交换;

NS2: 随机选择工序码的一个位置为交换点,将两端的序列直接旋转交换;

NS3: 随机选择工序码的一个位置,将其对应的数值从工序码中剔除,再随机插入剩余序列的任意位置;

NS4: 将原工序码整体进行倒序操作;

NS5: 任意选择工序码的一个位置数值 i 变为 $i+1(i \neq i(\max))$,同时任意选择工序码的一个位置数值 $i+1$ 变为 i .

4 多目标策略

本文通过计算种群个体的非支配等级^[12]和拥挤度的方法,对狼群中的最优解和最劣解进行保存,便于狼群在迭代过程中个体狼的位置更新.

4.1 计算个体非支配等级

对于种群规模为 $N(t)$ 的种群 P , t 为迭代次数. 具体操作步骤如下:

Step1 设 $i=1$;

Step2 对于所有的 $j=1,2,3 \dots N(t)$,且 $j \neq i$,基于个体的适应度函数比较个体 i 和个体 j 之间的非支配和支配关系. 如果所有个体 j 均不优于个体 i ,则将 i 判定为非支配个体;

Step3 令 $i=i+1$,返回Step1,直到找出所有的非支配个体.

通过Step1至Step3得到第一层的非支配解集. 然后,剔除第一层非支配解集对剩余个体重复Step1至Step3,得到第二层非支配解集...直到种群被分层. 种群规模随着迭代次数,不断进行更新操作.

4.2 计算个体拥挤度

拥挤度即为拥挤距离,是指单个Rank层中个体的密集程度. 计算个体拥挤度是为了得到同一等级Pareto解的优先级关系,拥挤度计算公式如下:

$$n_d = \sum_{v=1}^V \frac{f_v(i+1) - f_v(i-1)}{f_v^{\max} - f_v^{\min}} \quad (9)$$

其中: V 为目标函数的个数, $f_v(i-1)$ 和 $f_v(i+1)$ 为该个体排序之后前后两位目标函数值; f_v^{\max} 为某一目标函数的最大值; f_v^{\min} 为某一目标函数的最小值. 当个体 i 的拥挤度 n_{di} 大于个体 j 的拥挤度 n_{dj} , 则个体 i 占优.

5 IWPA解决MOFJSP步骤

Step1 种群初始化. 设置种群规模、迭代次数、探狼比例因子、探狼初始游走方向数、步长因子等参数. 计算个体非支配等级以及个体拥挤度后对个体进行排序, 选出适应度值最佳的狼为头狼, 并对头狼采用最佳序列机制进行调整 (在种群更新和迭代的过程中, 只要产生头狼, 就使用最佳序列更新机制, 更新头狼), 其余个体狼分别为探狼和猛狼;

Step2 种群迭代. 探狼位置更新; 头狼召唤猛狼, 猛狼位置更新; 然后计算猛狼和当前新头狼之间的距离, 判断猛狼是否转入猎物围攻行为; 围攻猎物之后, 使用物元分析法更新种群, 再次计算个体的非支配等级和拥挤距离, 选出最佳个体为种群新头狼;

Step3 判断是否满足要求, 若满足算法输出条件, 则输出对应的最优解以及适应度函数值; 若不满足, 返回Step2.

6 仿真测试及分析

仿真实验采用MATLAB 2018b实现, 运行环境: 处理器Core i5-8265U@1.60 GHz, Windows10操作系统. IWPA参数设置如下: 设置探狼比例因子为3, 探狼每一次游走的方向数最小值为4, 最大值为10; 步长因子为200; 改进狼群算法的种群规模和最大迭代次数等基本参数和比较算法保持一致.

为了验证IWPA的优势, 本文选择3种不同规模的算例, 使用不同文献中的算法进行求解比较. 3种算例分别为: Kacem1、Kacem2、Kacem3. 参考文献分别为: 混合人工蜂群算法 (Hybrid Artificial Bee Colony Algorithm, HTABC) [13]、状态转移算法 (State Transition Algorithm, STA) [14]、改进入侵杂草优化算法 (Improve Invasive Weed Optimization Algorithm, IIWO) [15]、多种邻域结构杂草优化算法 (Multiple Neighborhood Structures Weed Optimization Algorithm, MNSEO) [16]、狼群算法 (Wolf Colony Algorithm, WPA)、灰狼算法 (Grey Wolf Optimization, GWO). 对比结果如表4所示.

从“非劣解组数”“单目标最优解”“Pareto占优”三个层面对表4进行分析.

表 4 多种算法结果比较

算例	目标	HTABC		IIWO		STA		MNSEO	GWO		WPA		IWPA					
Kacem1 4个工件 5台机器	C_m	11	11	11	11	12	12	16	Kacem1: 无	12	11	12	12	11	11	12		
	W_t	32	34	33	32	33	32	32		33	32	31	32	31	33	32		
	W_m	10	9	10	10	8	8	8		8	11	11	10	11	10	8		
	单目标最优解	11,32,9		11,32,8		12,32,8			Kacem2: (15,75,13) (16,73,13) (17,78,11) (17,76,12) (18,75,12)	11,32,8		12,31,10		11,31,8				
Kacem2 8个工件 8台机器	C_m	14	15	15	15	15	16	16		15	16	16	16	14	15	15	16	16
	W_t	77	75	75	73	75	73	80		78	78	73	75	73	75	76	76	75
	W_m	12	15	12	12	13	13	12		12	11	12	11	13	11	12	12	13
	单目标最优解	14,75,12		15,73,12		16,73,12			15,78,11	16,73,11		14,73,11						
Kacem3 10个工件 7台机器	C_m	11	11	11		11	11		单目标最	11	11	11	11	11	11	11		
	W_t	61	61	61		63	62		优解:	61	66	61	62	63	61	62		
	W_m	11	11	11		10	11		(15,73,11)	11	10	12	11	10	12	11		
	单目标最优解	11,61,11				11,62,10			Kacem3: 无	11,61,10		11,61,11		11,61,10				

(1) 非劣解组数

算法求出的非劣解组数较多, 可以体现出算法有较好的求解性能. 针对Kacem1算例: IWPA可以求出3组非劣解, 和HTABC求解性能相同, 强于IIWO、STA、MNSEO、GWO、WPA; 针对Kacem2算例: IWPA可以求出5组非劣解, 和MNSEO求解性能相同, 强于HTABC、IIWO、STA、GWO、WPA; 针对Kacem3算例: IWPA可以求出3组非劣解, 和HTABC求解性能相同, 强于IIWO、STA、MNSEO、GWO、WPA.

(2) 单目标最优解

通过比较单目标最优解, 可以从整体性能上判断算法的优劣. 将每种算法求出的Pareto解集中的单个目标

的最优解进行取出合并,构成单目标最优解.针对Kacem1算例,IWPA求出非劣解的能力要强于HTABC、IIWO、STA、WPA、GWO;针对Kacem2算例,IWPA求出非劣解的能力要强于HTABC、IIWO、STA、MNSEO、GWO、WPA;针对Kacem3算例,IWPA求出非劣解的能力要强于HTABC、STA、WPA,和GWO能力大致相同.

(3) Pareto占优

比较不同算法之间的Pareto占优关系,对算法求解Pareto解集的能力进行综合判断.通过Pareto占优分析可知:针对Kacem1算例,IWPA求出非劣解的能力要强于WPA、GWO、IIWO,和STA大致相同,弱于HTABC;针对Kacem2算例,IWPA求出非劣解的能力要强于其他5种对比算法,和MNSEO大致相同;针对Kacem3算例,IWPA求出非劣解的能力要强于WPA、GWO、STA,弱于HTABC.针对Kacem3算例,选择一组非劣解集,使用IWPA画出相应的甘特图,如图1所示.

通过“非劣解个数”“单目标最优解”“Pareto占优”三个层面进行分析,可以得到:

(a) IWPA能够求出较多组数目的Pareto解集,求出的非支配解的数目越多,被选择为Pareto最优解的可能性就越大;

(b) IWPA有较好的单目标最优解,通过与其他智能优化算法所得结果进行比较,可以看出本文的IWPA具有一定的求解优势.针对Kacem1算例,IWPA求出的单目标最优解为(11,31,8),较其他算法有较强的求解优势;针对Kacem2算例,IWPA求出的单目标最优解为(14,73,11),IWPA求出非劣解的能力要强于HTABC、IIWO、STA、MNSEO、GWO、WPA;针对Kacem3算例,IWPA求出非劣解的能力要强于HTABC、IIWO、STA、MNSEO、GWO、WPA;IWPA在Pareto占优层面,也具有较弱的寻优优势,能够向组合后的单目标最优解提供较多的选择.

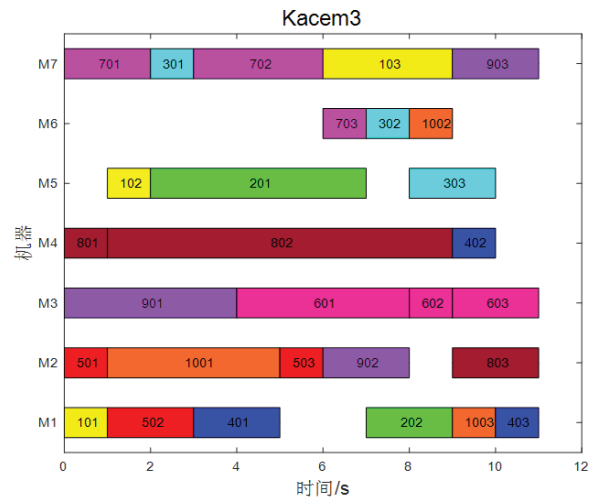


图 1 Kacem3调度甘特图

7 结论

本文基于狼群算法的基本结构和寻优优势,提出一种将量子粒子群算法的重要参数和狼群算法融合的混合优化算法对多目标柔性作业车间调度问题进行求解.

(1) 新型混合优化算法使寻优种群在一定的搜索范围之内,以较快的收敛速度朝着全局最优个体不断靠拢,算法求解全局最优解的能力得以增强;

(2) 采用高斯分布的概率密度函数对种群进行初始化,提高初始种群的质量和多样性;通过最佳序列调整机制,不断更新最优解的同时也提高了算法的收敛速度;

(3) 物元分析法丰富了种群的更新策略,有利于种群的迭代进化.

通过以上分析,可知本文所提出的混合优化算法对求解MOFJSP具有一定的合理性和优势.笔者将在本文研究基础之上,进一步探究混合狼群算法应用于不确定因素影响、动态重调度等更加复杂的车间调度问题.

参考文献:

- [1] 陈魁, 毕利. 改进粒子群算法在考虑运输时间下的FJSP研究[J/OL]. 系统仿真学报, 2021, 33(4): 845-853.
- [2] 景志强, 王兆辉, 高琦. 混合NSGA-II算法求解多目标柔性作业车间调度问题[J]. 组合机床与自动化加工技术, 2019(7): 138-140+145.
- [3] HUANG X, GUAN Z, YANG L. An effective hybrid algorithm for multi-objective flexible job-shop scheduling problem[J]. Advances in Mechanical Engineering, 2018, 10(9): 1-14.
- [4] 朱光宇, 冯子超, 杨志锋. 灰熵并行分析引导PSO求解多目标优化问题[J]. 系统工程与电子技术, 2014, 36(11): 2233-2238.
- [5] 朱伟, 赵水晶. 考虑工件移动和成本的多目标柔性作业车间调度问题优化研究[J]. 现代制造工程, 2019, 7: 104-110+95.