

3-22-2023

Flexible Job-Shop Scheduling Problem Based on Improved Wolf Pack Algorithm

Chaoyang Zhang

School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China; 754690654@qq.com

Liping Xu

School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China; xlpzz@163.com

Jian Li

School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China;

Yihao Zhao

School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China;

See next page for additional authors

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Flexible Job-Shop Scheduling Problem Based on Improved Wolf Pack Algorithm

Abstract

Abstract: An improved wolf pack algorithm is proposed for solving multi-objective scheduling optimization for flexible job shop problems. A multi-objective flexible job shop scheduling model is developed with the maximum completion time of the workpiece and the energy consumption of the machine as the optimization goals. *An improved wolf pack algorithm is proposed for solving the shortcomings that traditional wolf pack algorithm is easy to fall into the local optimization. Through improving the intelligent behavior of the wolf pack algorithm, individual codes are designed from the two levels of job's process and machine, and POX(precedence operation crossover) cross operation is introduced to ensure the legality of the solution and improve the search ability of the algorithm.* The effectiveness of the improved wolf pack algorithm is verified through comparative experiments on two workshop examples. Experimental results show, the improved wolf pack algorithm not only has good global search ability, but also has an improved optimization ability compared with other algorithms. It can provide new solutions for the manufacturing industry to improve production efficiency.

Keywords

improved wolf pack algorithm, job shop scheduling, energy consuming, multi-objective optimization

Authors

Chaoyang Zhang, Liping Xu, Jian Li, Yihao Zhao, and Kui He

Recommended Citation

Chaoyang Zhang, Liping Xu, Jian Li, Yihao Zhao, Kui He. Flexible Job-Shop Scheduling Problem Based on Improved Wolf Pack Algorithm[J]. Journal of System Simulation, 2023, 35(3): 534-543.

基于改进狼群算法的柔性作业车间调度研究

张朝阳, 徐莉萍*, 李健, 赵义豪, 何奎

(河南科技大学 机电工程学院, 河南 洛阳 471000)

摘要: 将一种改进的狼群算法用于解决柔性作业车间多目标调度优化的难题。以工件的最大完工时间和机器能耗为优化目标建立了多目标柔性作业车间调度模型; 针对传统狼群算法容易陷入局部最优的缺点提出了一种改进的狼群算法, 通过对狼群算法智能行为的改进, 从工序和机器 2 个层面设计个体编码, 引入 POX (precedence operation crossover) 交叉操作, 保证解的合法性, 提高算法的全局搜索能力; 通过对 2 个车间实例的对比实验来验证改进狼群算法的有效性。实验结果表明, 提出的改进狼群算法不仅具有良好的全局搜索能力, 寻优能力较其他算法也有所提升, 能够为制造业提高生产效率提供新的解决思路。

关键词: 改进狼群算法; 车间调度; 能耗; 多目标优化

中图分类号: TP278; TP391.9 文献标志码: A 文章编号: 1004-731X(2023)03-0534-10

DOI: 10.16182/j.issn1004731x.joss.21-1019

引用格式: 张朝阳, 徐莉萍, 李健, 等. 基于改进狼群算法的柔性作业车间调度研究[J]. 系统仿真学报, 2023, 35(3): 534-543.

Reference format: Zhang Chaoyang, Xu Liping, Li Jian, et al. Flexible Job-Shop Scheduling Problem Based on Improved Wolf Pack Algorithm[J]. Journal of System Simulation, 2023, 35(3): 534-543.

Flexible Job-Shop Scheduling Problem Based on Improved Wolf Pack Algorithm

Zhang Chaoyang, Xu Liping*, Li Jian, Zhao Yihao, He Kui

(School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China)

Abstract: An improved wolf pack algorithm is proposed for solving multi-objective scheduling optimization for flexible job shop problems. A multi-objective flexible job shop scheduling model is developed with the maximum completion time of the workpiece and the energy consumption of the machine as the optimization goals. An improved wolf pack algorithm is proposed for solving the shortcomings that traditional wolf pack algorithm is easy to fall into the local optimization. Through improving the intelligent behavior of the wolf pack algorithm, individual codes are designed from the two levels of job's process and machine, and POX(precedence operation crossover) cross operation is introduced to ensure the legality of the solution and improve the search ability of the algorithm. The effectiveness of the improved wolf pack algorithm is verified through comparative experiments on two workshop examples. Experimental results show, the improved wolf pack algorithm not only has good global search ability, but also has an improved optimization ability compared with other algorithms. It can provide new solutions for the manufacturing industry to improve production efficiency.

Keywords: improved wolf pack algorithm; job shop scheduling; energy consuming; multi-objective optimization

收稿日期: 2021-10-04 修回日期: 2021-12-07

基金项目: 国家重点研发计划(2018YFB1701205)

第一作者: 张朝阳(1994-), 男, 硕士生, 研究方向为智能产线优化。E-mail: 754690654@qq.com

通讯作者: 徐莉萍(1965-), 女, 副教授, 硕士, 研究方向为机电液一体化。E-mail: xlpzz@163.com

0 引言

随着工业 4.0 和中国制造 2025 的提出, 制造业正向智能化方向转型。生产调度作为生产运作过程中的重要环节, 已成为制造业快速响应定制化需求的有效途径。作业车间调度问题是生产调度中具有重要意义的基础问题^[1]。随着现代生产过程向多品种、小批量、定制化方向的转变, 柔性作业车间调度问题作为作业车间调度问题的扩展形式, 更符合现代生产的要求。柔性作业车间调度 (flexible job shop scheduling problem, FJSP) 较作业车间调度更具灵活性和复杂性, 如何优化柔性作业车间调度方案用以提高机器利用率、缩短生产时间、降低生产能耗, 成为了当代相关领域研究的热点和难点。

过去几十年中, 专家学者们针对 FJSP 问题, 分别从确定性算法和近似性算法进行了深入研究^[2]。由于其非确定性问题 (nondeterministic problem, NP) 难的本质^[3], 随着研究的不断深入, 越来越多的研究工作趋向于采用具有不确定性的元启发式算法和智能优化算法进行求解。如文献[4]以最大完工时间为目标解决 FJSP 问题, 提出混合差分进化算法, 通过增强局部搜索来提高整个算法的效率; 文献[5]提出混合整数线性规划模型, 针对 FJSP 问题, 采用停机策略减小最大完工时间和总能耗, 并采用遗传算法进行实例研究, 在保证最大生产力的同时, 显著降低能耗; 文献[6]针对 FJSP 调度问题, 将其分为调度和重调度两个层面来最小化最大完工时间, 并采用两阶段人工蜂群算法进行求解; 文献[7]针对 FJSP 问题, 提出分层多空间竞争分布式遗传算法, 上层用于确定解的搜索空间, 给定解空间的最优解在下层确定, 来实现最大完工时间最小化; 文献[8]研究了有关绿色调度的 FJSP 问题, 以最大程度地减少能耗成本和拖期成本, 并对原始的猫群优化算法进行了修改, 提出了双种群离散猫群优化算法; 文献[9]针对 FJSP 问题, 设置一个能耗阈值, 研究总

能耗在不超过给定阈值的约束下, 最小化完工时间和总拖期, 提出了一种两阶段元启发式算法; 文献[10]以改进遗传算法并利用精英保留策略提高求解质量; 文献[11]将入侵杂草算法进行改进, 用于多目标 FJSP 优化; 文献[12]针对 FJSP 问题, 采用细菌觅食优化算法, 对趋化、复制和迁移/驱散操作进行改进, 改善算法的全局开发能力和局部搜索能力。目前对于 FJSP 问题, 传统的元启发式算法在求解能力上有所不足, 在寻优能力上仍有一定的局限性。因此寻找更好的算法用于提高车间调度效率依旧是当今的热点之一。

狼群算法 (wolf pack algorithm, WPA) 是近年来提出的一种群体智能算法, 该算法具有较强的全局收敛性和计算鲁棒性^[13]。目前 WPA 在多选择背包问题、旅行商问题、资源调度问题上取得了不错的效果^[14-16]。文献[17-18]采用狼群算法以最大完工时间为目标对车间调度问题进行了单目标优化研究。

WPA 在车间调度领域的研究尚不成熟, 其原因在于传统的 WPA 算法因其参数过多容易出现局部最优, 收敛较差等缺陷。本文一方面将 WPA 的应用对象扩展到多目标 FJSP 问题上; 另一方面, 对 WPA 算法中的 3 种智能行为进行了改进, 在游走过程中引入工序和机器双层游走方式, 在召唤行为中引入 POX (precedence operation crossover) 交叉操作, 确保优良个体基因能被保留下来, 在围攻行为中设置个体自交叉的方式, 提高算法的全局搜索能力。通过相关实验证明改进后的 WPA 在多目标车间调度优化当中有效缩短了最大完工时间和机器能耗, 最后得到了对应的调度甘特图。

1 问题描述及建模

1.1 问题描述

柔性作业车间可以定义为: n 个工件在 m 台机器上进行加工, 每个工件有多道工序, 且每道工序可以在多台机器上加工。

本文以工件的最大完工时间和机器总能耗为目标, 通过合理安排工件工序并分配机器进行加工, 来实现能耗和最大完工时间性能指标达到整体最优。在建立模型之前, 做出以下假设:

- (1) 每道工序只能在一台机器上被加工一次;
- (2) 同一工件的各道工序存在工艺约束;
- (3) 工件在加工过程中不能中断;
- (4) 同一时间, 每台机器只能加工一个工件;
- (5) 0时刻工件和机器已准备就绪;
- (6) 不考虑工件的运输时间及机器的换刀和装夹时间。

1.2 相关符号说明

为建立最大完工时间和能耗的调度模型, 给出以下符号说明, 如表 1 所示。

表 1 符号说明

符号	定义
$N=\{1, 2, \dots, i, \dots, n\}$	工件
$J=\{1, 2, \dots, j, \dots, q\}$	工序
$M=\{1, 2, \dots, k, \dots, m\}$	机器
$O_{i,j}$	工件 i 的第 j 道工序
$S_{i,j,k}$	$O_{i,j}$ 的开始加工时间
$F_{i,j,k}$	$O_{i,j}$ 的结束加工时间
C_i	工件 i 的完工时间
$T_{i,j,k}$	$O_{i,j}$ 在机器 k 上的加工时间
$U_{i,j,k}$	整数变量, 取 0 或者 1, 如果 $O_{i,j}$ 在机器 k 上加工, 则为 1; 否则为 0
P_c^k	机器 k 的加工功率
P_{idle}^k	机器 k 的待机功率
T_c^k	机器 k 的加工时间
T_{idle}^k	机器 k 的待机时间
E_k	机器 k 的总能耗
E_c^k	机器 k 的加工能耗
E_{idle}^k	机器 k 的待机能耗

1.3 模型建立

假设有 n 个工件在 m 台机器上进行加工。机器在运行过程中, 一般分为启动、空载、加工和关机 4 种状态。机器开始加工到完成加工期间,

多次启停会减少机器寿命, 故一般机器在开机之后保持空载或加工状态, 直到完成加工。因此, 启停操作只有一次, 其能耗值较小, 可以忽略不计。

- (1) 最大完工时间

最大完工时间是所有工件的完工时间。

$$f_1 = \max_{1 \leq i \leq n} C_i \quad (1)$$

$$C_i = F_{i,q} \quad (2)$$

式中: $F_{i,q}$ 为工件 i 最后一道工序的完工时间。

- (2) 机器总能耗

机器的总能耗(E)分为加工能耗(E_c)和待机能耗(E_{idle}), 加工能耗是机器的加工功率和加工时间的乘积, 同样, 待机能耗取决于机器的待机功率和待机时间。由此可得总能耗公式为

$$f_2 = E = \sum_{k=1}^m E_k = \sum_{k=1}^m (E_c^k + E_{idle}^k) = \sum_{k=1}^m (P_c^k \cdot T_c^k + P_{idle}^k \cdot T_{idle}^k) \quad (3)$$

机器能耗和最大完工时间的具体数学模型为

$$\begin{cases} \min f_1 = \max_{1 \leq i \leq n} C_i \\ \min f_2 = E \end{cases} \quad (4)$$

约束条件如下:

- (1) 工件的下道工序加工要在上道工序结束之后:

$$S_{i,j} \geq F_{i,j-1} \quad (5)$$

- (2) 每道工序只能被加工一次:

$$\sum_{k=1}^m U_{i,j,k} = 1, i \in N \quad (6)$$

- (3) 机器在加工当前工序之后才能加工下道工序:

$$F_{i,k} \leq S_{i',k}, i, i' \in N, k \in M \quad (7)$$

上述数学模型是双目标优化模型, 本文首先对目标函数 f_1 和 f_2 进行归一化处理, 之后采用加权法对其进行求解。

归一化公式为

$$f_i^* = \frac{f_i - \mu_i}{\sigma_i} \quad (8)$$

式中: μ_i 为均值; σ_i 为方差, $i=1, 2$ 。

加权之后的目标函数为

$$f = a_1 f_1^* + a_2 f_2^* \quad (9)$$

式中: a_1 和 a_2 分别为最大完工时间和能耗的权重, 满足 $a_1 + a_2 = 1$; f_1^* 和 f_2^* 分别为 f_1 和 f_2 的归一化值。

2 狼群算法

狼群算法是受到自然界狼群围捕猎物现象启发而得出的群体智能优化算法, 包括三种智能行为(游走行为、召唤行为和围攻行为)、“头狼产生机制和狼群更新机制。狼群算法流程如下:

步骤1: 初始化操作。计算各人工狼的目标函数值, 选取最优的人工狼作为头狼。

步骤2: 游走行为。选取部分人工狼为探狼执行游走行为, 直到某探狼的位置所得到的目标函数值大于头狼的目标函数值或者达到最大游走次数。

步骤3: 召唤行为。选取剩余的人工狼作为猛狼向头狼位置迅速移动。若移动途中, 其位置的目标函数值大于头狼, 则该猛狼替换为头狼从新发起召唤行为, 直到所有猛狼都在头狼的感知距离范围内。

步骤4: 围攻行为。探狼和猛狼一起向猎物发起围攻。若其目标函数值优于原始位置, 则更新其位置。

步骤5: 淘汰机制。按照人工狼的目标函数值的大小进行排序, 剔除掉最差的 R 匹人工狼, 同时初始化 R 匹人工狼补充到种群中去。

步骤6: 判断是否达到最大迭代次数, 若达到则输出头狼的位置, 否则转到步骤2。

3 求解FJSP的改进狼群算法

狼群算法最初用于连续函数优化问题, 在求解组合优化问题时, 易陷入局部最优, 过早收敛, 得不到最优函数值。针对FJSP问题特征, 对狼群

算法三种智能行为进行改进。以扩大搜索范围获得全局最优解。

3.1 编码方式

因涉及工序安排和机器选择两个方面, 在此采用工序和机器的双层编码方式, 并且使工序层和机器层编码长度相等, 同时位置对应。编码方式如表2所示。

表2 编码段

Table 2 Code segment

工序层	2	3	2	1	3	2	3	1	1
机器层	1	1	1	3	3	1	2	2	3

表2中第一行代表工序, 其中数字代表其工件名, 其出现的次数代表工件的工序, 比如“1”表示工件1, 第一次出现的“1”表示工件1的第一道工序, 第二次出现的“1”表示工件1的第二道工序, 以此类推; 第二行是加工工序对机器的选择问题, 每个工序对应一个机器集, 每个数字表示其机器集的位置索引。如工序 $O_{2,1}$ 的加工机器集为 $[M_2, M_4]$ (M_2 和 M_4 分别表示机器2和机器4), 1就表示其加工机器为机器集的第一个位置, 即是 M_2 , 表示工序 $O_{2,1}$ 在机器2上加工。

3.2 定义

狼群的狩猎空间定义为 $N \cdot D$, N 为人工狼的数量, D 为工序层和机器层编码长度之和。在这里, 用 $X_i (i=1, 2, \dots, N)$ 表示人工狼的位置, 则人工狼 i 的位置可以表示为 $X_i = [x_{i1}, x_{i2}, \dots, x_{id}, \dots, x_{iD}]$ (其中 $x_{i1}, x_{i2}, \dots, x_{iD}$ 表示工序层和机器层编码), 人工狼的气味浓度值定义为 Y_i 。在这里, 气味浓度值即为适应度函数值, 也就是本文所求的目标函数值。

3.3 种群初始化

本文采用3个规则来生成初始解: 随机生成法、最小加工能耗法、最小时间选择法。采用3种规则生成的种群规模分别为40%、30%、

30%，其目的是提高种群初始解的质量。

3.4 智能行为设计

(1) 头狼产生机制。根据目标函数计算各人工狼的气味浓度值。人工狼 i 的气味浓度值为 Y_i ，则头狼 $Y_{lead} = \min(Y_i), i = 1, 2, \dots, N$ ，若同时存在多个最小值，则进行随机选择。与头狼对应的解和气味浓度值分别为 X_{lead} 和 Y_{lead} 。

(2) 游走行为。选取除头狼外的目标函数值最优的 S_num 匹人工狼作为探狼。记探狼 i 的位置为 $X_i = [x_{i1}, x_{i2}, \dots, x_{id}, \dots, x_{iD}]$ ，当前气味浓度值为 Y_i 。探狼向 h 个方向进行游走，记探狼 i 向第 p 个方向游走后的气味浓度值为 Y_{ip} ，如果 $Y_{ip} > Y_i$ ，则更新其位置，否则不变；如果 $Y_i > Y_{lead}$ ，更换该探狼为头狼。重复以上游走行为，直到满足 $Y_i > Y_{lead}$ 或者游走次数 T 达到最大游走次数 T_{max} 。

对于游走行为，进行如下改进：探狼 i 采取编码游走方式，定义游走步长 $step_{a1}$ 为该探狼游走的个体位置向量数。游走行为可以分为工序游走和机器游走两种行为。工序游走行为如图 1 所示，该工序编码为 [2, 3, 2, 1, 3, 2, 3, 1, 1]，设工序游走步长 $step_{a1} = 4$ ，随机选择 4 个包含不同工件的位置为 (1, 3, 7, 9)，其编码为 [2, 2, 3, 1]，对其进行随机排序，排序结果为 [3, 2, 1, 2]，则游走后的工序编码为 [3, 3, 2, 1, 3, 2, 1, 1, 2]。

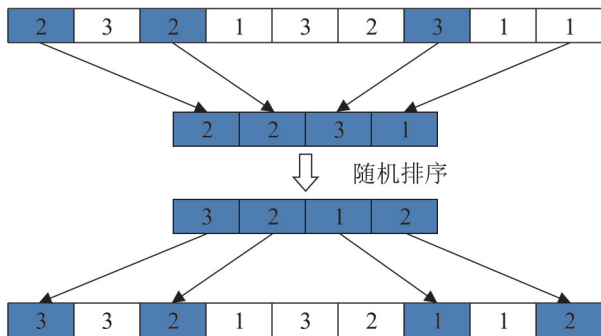


图 1 工序游走行为
Fig. 1 Process wandering behavior

对机器编码进行游走操作，假设机器游走步长 $step_{a2} = 1$ ，任选一个加工机器集不少于 2 台的机

器的工序，在其机器集里进行随机选择，如图 2 所示，随机选择工序 $O_{3,2}$ 对应的机器编码为 3，对应的机器号为 M_5 ，对其进行随机选择，其结果为 M_1 。

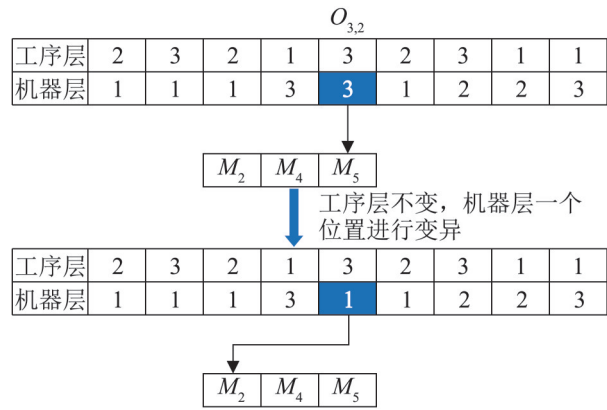


图 2 机器游走行为
Fig. 2 Machine wandering behavior

(3) 召唤行为。将剩余的 M_sum 匹人工狼看作猛狼，头狼召唤猛狼向猎物中心进行移动。其中 $M_sum = N - S_sum - 1$ 。头狼召唤周围的猛狼向猎物方向移动。

对于召唤行为，进行如下改进：采用 POX (precedence operation crossover) 交叉^[19]方式来体现头狼对人工狼的召唤，能够更好的保留头狼的优秀特征。选取某个人工狼 X_i 和头狼 X_{lead} 进行工序层的交叉，机器层跟随工序层变化。如图 3 所示，首先针对 3 个工件可以分为 2 个集合，集合中的数字代表工件名，得出 $J_1 = \{2\}$ ， $J_2 = \{1, 3\}$ ，选择包含工件个数最少的子集 J_1 ，复制 X_i 中包含 J_1 的工件编码到 X'_i 的相同编码位置中，复制 X_{lead} 中包含 J_1 的工件编码到 X'_{lead} 相同编码位置中；复制 X_i 中包含 J_2 的工件编码，按顺序插入到 X'_{lead} 中，同理，复制 X_{lead} 中包含 J_2 的工件编码，按顺序插入到 X'_i 中，其机器编码与工序编码进行相对应变化。

(4) 围攻行为。头狼联合猛狼和探狼向猎物发起围攻，在这里头狼离猎物最近，可以把头狼 X_{lead} 的位置视为猎物位置，若实施围攻行为后人工狼气味浓度优于其原位置，则更新此人工狼的

位置, 否则不变。围攻行为仅针对工序编码, 机器编码进行相应变换即可。这里和改进的游走行为类似, 围攻步长为 $step_c$, 定义为整数。假设人工狼 X_i 的编码为 [1, 3, 2, 2, 3, 1, 2, 1, 3], 个体编码数即为工序编码长度9, 围攻步长转换为个体编码数, 其取值范围为 [0, 9] 的整数, 这里围攻步长应选择合理, 围攻步长太大容易跳出最优解的范围, 一般取值为个体编码数的 1/3~1/2。

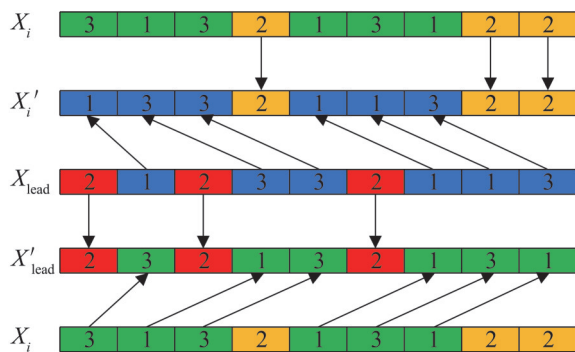


图3 POX交叉
Fig. 3 POX cross

(5) 更新机制。在进行围攻行为之后, 人工狼按气味浓度值 Y_i 从大到小的顺序排列, 去掉掉气味浓度值最低(即为目标函数值较大)的 R 匹人工狼, 同时生成 R 匹人工狼。一般取 $R \in [M/2\beta, M/\beta]$, β 为种群更新比例因子。

4 实验仿真分析

采用 FJSP 的 2 个实际算例进行仿真试验以测试算法有效性。改进的狼群算法具体参数设置如表 3 所示。

表3 算法参数
Table 3 Algorithmic parameters

参数名	数值
种群规模	100
迭代次数	200
探狼比例因子	0.4
最大游走次数	10
工序游走步长	4
机器游走步长	2
更新比例因子	0.3

本文以文献[10]和文献[20]的改进遗传算法作为对比, 其遗传算法主要步骤为:

步骤1: 种群初始化;

步骤2: 根据个体权重比计算适应度值, 并进行排序;

步骤3: 选择、交叉和变异操作;

步骤4: 是否求得最优解, 否则回到步骤2。

4.1 实例1

实例1是 5×5 的实例^[10], 具体工艺参数如表4所示。

表4 5个工件在5台机器上的加工工艺
Table 4 Processing technology of 5 workpieces on 5 machines

工件	工序	机器				
		M_1	M_2	M_3	M_4	M_5
J_1	O_{11}	5	4	7	4	10
	O_{12}	4	4	10	3	9
	O_{13}	5	8	3	6	10
J_2	O_{21}	8	9	9	9	4
	O_{22}	3	2	2	1	7
	O_{23}	8	2	4	7	8
J_3	O_{31}	7	2	1	9	9
	O_{32}	8	7	7	8	2
	O_{33}	4	6	1	7	1
J_4	O_{41}	1	8	7	9	8
	O_{42}	7	8	3	4	10
	O_{43}	8	3	1	6	6
J_5	O_{51}	3	7	7	10	6
	O_{52}	2	10	9	5	10
	O_{53}	2	2	5	9	5
加工功率 P_i /kW		2	1.8	1.6	2.4	3.8
待机功率 P_{idle} /kW		0.5	0.6	0.3	0.4	0.5

采用表4所示的加工工艺, 对改进的狼群算法和传统遗传算法进行对比, 取权重 $\alpha_1 = 1$ 时, 即只考虑最大完工时间的单目标进行优化, 求解最小的完工时间。从图4可以看出, 改进的狼群算法在第30代左右完成收敛, 而遗传算法要在60代以后才完成收敛, 并且其最优解要次于改进狼群算法, 所以改进狼群算法在收敛速度和寻优能力上比遗传算法要好。

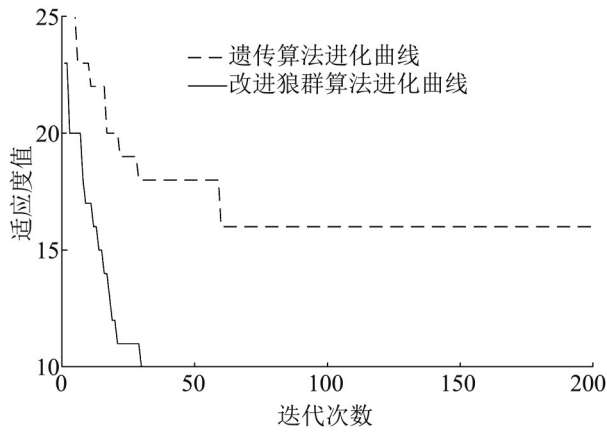


图4 算法进化曲线对比结果
Fig. 4 Algorithmic evolution curves contrast results

当 $\alpha_1=0$ 时，即只考虑机器总能耗，此时转为考虑能耗的单目标优化问题，进化曲线如图 5 所示。由图 5 可知，种群在 40 代左右收敛，最优解在 30 代左右得到。

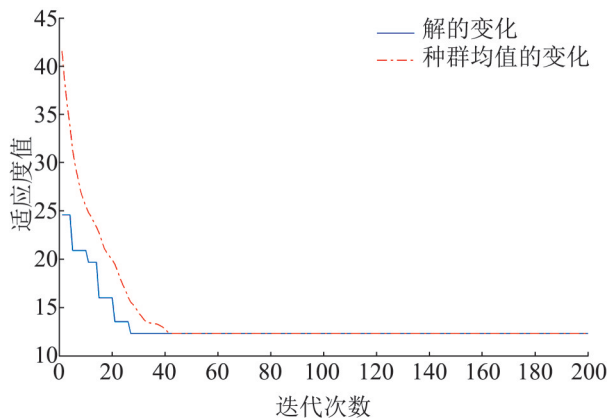


图5 迭代次数
Fig. 5 Number of iterations

图 6 为 $\alpha_1=0$ 的调度甘特图，从图中可知，最大完工时间为 10，能耗值为 80.0。

表 5 是根据不同权重组合，得到的不同调度方案。从实验中得知，适应度值整体上呈现增大的趋势。实际生产中，企业可以根据时间和经济 2 个方面进行综合考量，根据实际需求对最大完工时间和能耗 2 个目标的权重系数进行调整。

改进狼群算法和遗传算法的对比结果如表 6 所示，相较文献[10]采用改进的遗传算法对最大完

工时间和能耗进行优化求解， $\alpha_1=1$ 时，为最大完工时间最小化，最大完工时间从 16 降低为 10，降低了 37.5%； $\alpha_1=0$ 时，为能耗值最小化，能耗值从 95.2 降低为 74，降低了 22.3%。

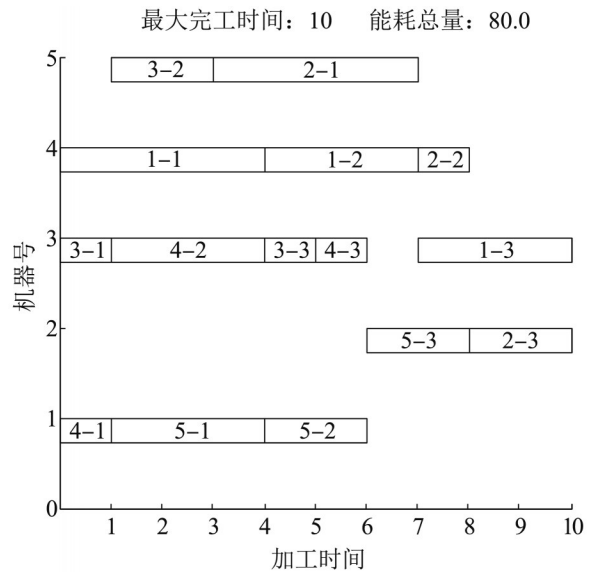


图6 调度甘特图
Fig. 6 Scheduling Gantt Chart

表 5 不同权重组合下的数据
Table 5 Data under different weight combinations

实验序号	α_1	α_2	最大完工时间	能耗值	适应度值
1	1	0	10	80.0	12.300
2	0.9	0.1	10	76.3	12.410
3	0.8	0.2	11	86.0	13.834
4	0.7	0.3	10	79.8	12.800
5	0.6	0.4	10	76.3	12.720
6	0.5	0.5	11	78.3	13.620
7	0.4	0.6	10	76.3	12.931
8	0.3	0.7	10	76.3	13.037
9	0.2	0.8	11	74.4	13.120
10	0.1	0.9	11	74.4	13.071
11	0	1	12	74.0	12.950

表 6 实例 1 对比结果
Table 6 Comparison results of instance 1

算法	$\alpha_1=1$		$\alpha_1=0$	
	最大完工时间	能耗值	最大完工时间	能耗值
改进遗传算法	16	109	31	95.2
改进狼群算法	10	80	12	74

4.2 实例2

实例2数据^[20]为6个工件在6台机器上进行加工, 在这里, 需保证同一机器的加工功率在加工不同工件时一致, 取其可加工工件的平均值。加工工艺数据见表7。

表7 6个工件在6台机器上的加工工艺
Table 7 Processing technology of 6 workpieces on 6 machines

工件	工序	机器					
		M_1	M_2	M_3	M_4	M_5	M_6
J_1	O_{11}	10	15		14		14
	O_{12}		6	4	16	15	
	O_{13}		5			16	8
	O_{14}	12		13	13		
J_2	O_{21}	15		6	16.5	11	
	O_{22}		15	10	7		12
	O_{23}	5		16	10	14	
	O_{24}		10				
J_3	O_{31}	14	15	6	5	4	
	O_{32}		5	6		16	
	O_{33}	5	8		11		15
	O_{34}		6	17	14	12	
J_4	O_{41}	20		19	13	15	
	O_{42}		10	7	14	7	15
	O_{43}	4	8				16
	O_{44}	9		6		6	
J_5	O_{51}		6		7	12	8
	O_{52}	8		12	16		6
	O_{53}	13	12			16	8
	O_{54}		4	6	5	12	
J_6	O_{61}		11			7	8
	O_{62}			8	12		6
	O_{63}	10	5		13	6	
	O_{64}	16		8			12
加工功率 P_c/kW	2.66	2.23	1.83	2.56	2.35	1.35	
待机功率 $P_{\text{idle}}/\text{kW}$	0.50	0.40	0.35	0.41	0.36	0.25	

通过设置不同权重 α_1 和 α_2 , 每个权重组合下进行多次仿真, 取其出现次数多的一组解, 得到实验数据如表8所示。

其完工时间和能耗值随着权重的变化比较明

显。实例2中改进狼群算法和文献20的传统遗传算法对比结果如表9所示, 相较文献[20]采用的传统遗传算法, 在 $\alpha_1=1$ 时, 最大完工时间从51降低为37, 缩短了27.5%。

表8中的实验1为最大完工时间最小时的调度甘特图, 如图7所示。

表8 不同权重组合下的数据
Table 8 Data under different weight combinations

实验序号	α_1	α_2	最大完工时间	能耗值
1	1	0	37	424.8
2	0.9	0.1	39	420.3
3	0.8	0.2	40	382.1
4	0.5	0.5	44	375.5
5	0.2	0.8	41	386.8
6	0.1	0.9	44	375.3
7	0	1	54	370.7

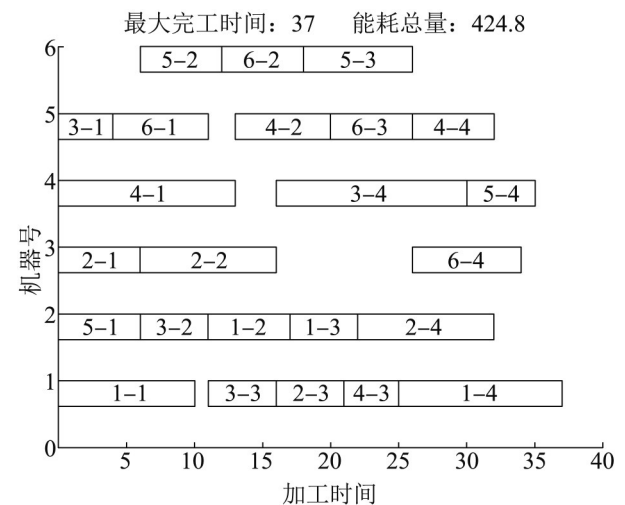


图7 完工时间最小时调度甘特图

Fig. 7 Scheduling Gantt chart with minimum completion time

由图7可知, 为了缩小最大完工时间, 不考虑机器的能耗问题, 工件尽可能的平均分布在各个机器上进行加工。

表7中的实验7为求最小能耗值, 调度甘特图如图8所示。由图8可知, 为了保证最小能耗, 工件尽可能的在加工时长和加工功率较小的机器上加工, 因为大多数工件在机器4和5上的加工时长都较长, 其能耗对应就比较大, 所以工件选择较少。

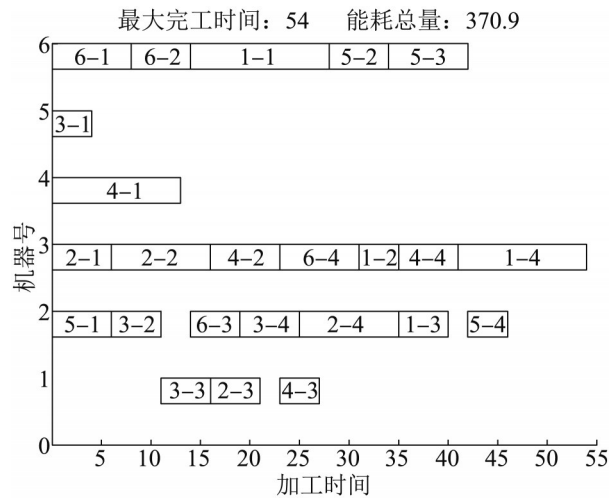


图8 能耗最小时调度甘特图

Fig. 8 Scheduling Gantt chart with minimum energy consumption

4.3 大规模实例测试

为测试改进狼群算法在大规模工件加工上的有效性, 本文基于实例1和实例2的加工数据, 在加工机器个数不变的情况下, 扩大加工工件的规模, 每种工件加工的个数从1件扩大至10件, 将改进狼群算法(IWPA)与遗传算法(GA)的测试结果进行比较。每种算法各运行10次, 并对结果进行归一化处理, 式(10)和式(11)为归一化公式, 设定权重值 $\alpha_1=0.5$, $\alpha_2=0.5$, 并选取最优值, 对比结果如表9所示。

表9 实例2对比结果

Table 9 Comparison results of instance 2

算法	$\alpha_1=1$		$\alpha_1=0$	
	最大完工时间	能耗值	最大完工时间	能耗值
改进遗传算法	51	—	61	—
改进狼群算法	37	424.8	54	370.7

$$f_i^* = \frac{f_i - f_i^{\min}}{f_i^{\max} - f_i^{\min}} \quad (10)$$

式中: f_i^{\max} 和 f_i^{\min} 分别为10次结果中的最大值和最小值。

$$f = \alpha_1 f_1^* + \alpha_2 f_2^* \quad (11)$$

从表10可以看出, IWPA在最大完工时间 f_1

和能耗值 f_2 上的效果均优于GA, 完工时间和能耗值均有所降低。对比实验证明, 本文提出的IWPA同样适用于大规模工件加工。

表10 大规模实例对比结果

Table 10 Comparison results of large-scale instances

目标参数	50×5		60×6	
	IWPA	GA	IWPA	GA
f_1 完工时间	77	87	346	358
f_2 能耗值	818.1	832.9	3 606.2	3 746.4
f 归一化值	0.15	0.24	0.25	0.28

5 结论

本文针对柔性作业车间调度问题, 从最大完工时间和机器能耗两个方面进行优化调度, 提出了一种改进的狼群算法以寻找最优工件排序和机器选择, 使最大完工时间最小化, 能耗最低。该优化算法, 采用双层编码机制来解决工件排序和机器选择问题; 通过重新定义狼群算法的3种智能行为, 提高全局搜索能力, 并引入POX交叉操作, 以保证所得解的合法性, 解决了传统狼群算法在解决车间调度问题容易陷入局部最优的问题。利用本文提出的改进狼群算法求得的最大完工时间和机器能耗大幅降低, 相较于文献10提出的改进的遗传算法最大完工时间和能耗分别缩短了33.3%和22.2%, 较文献20提出的多层编码遗传算法最大完工时间缩短了27.5%, 同时在大规模实例上有一定效果, 验证了本文提出的改进狼群算法的有效性。由于本文没有考虑AGV小车设备的运输和能耗, 所建模型具有一定的局限性, 后续的研究工作将在现有的基础上, 将AGV小车的运输时间和运输能耗加入到研究对象中去, 以进一步完善调度模型。

参考文献:

- [1] Johnson S M. Optimal Two-and Three-Stage Production Schedules with Setup Times Included[J]. Naval Research Logistics Quarterly (S0894-069X), 1954, 12(1): 336-353.
- [2] Chaudhry, Imran, Ali, et al. A Research Survey: Review of Flexible Job Shop Scheduling Techniques[J]. International

- Transactions in Operational Research (S0969-6016), 2005, 23(3): 551-591.
- [3] Pezzella F, Morganti G, Ciaschetti G. A Genetic Algorithm for the Flexible Job-Shop Scheduling Problem [J]. Computers & Operations Research (S0305-0548), 2008, 35(10): 3202-3212.
- [4] Yuan Y, Xu H. Flexible Job Shop Scheduling Using Hybrid Differential Evolution Algorithms[J]. Computers & Industrial Engineerin (S0360-8352), 2013, 65(2): 246-260.
- [5] Ham A, Park M J, Kim K M, et al. Energy-Aware Flexible Job Shop Scheduling Using Mixed Integer Programming and Constraint Programming[J]. Mathematical Problems in Engineering (S1024-123X), 2021: 1-12.
- [6] Gao K Z, Suganthan P N, Chua T J, et al. A Two-stage Artificial Bee Colony Algorithm Scheduling Flexible Job-Shop Scheduling Problem with New Job Insertion[J]. Expert Systems with Applications (S0957-4174), 2015, 42(21): 7652-7663.
- [7] Ishikawa S, Kubota R, Horio K. Effective Hierarchical Optimization by a Hierarchical Multi-space Competitive Genetic Algorithm for the Flexible Job-Shop Scheduling Problem[J]. Expert Systems with Applications (S0957-4174), 2015, 42(24): 9434-9440.
- [8] Jiang T, Deng G. Optimizing the Low-Carbon Flexible Job Shop Scheduling Problem Considering Energy Consumption[J]. IEEE Access (S2169-3536), 2018, 6: 99.
- [9] Lei D M, Li M, Wang L. A Two-Phase Meta-Heuristic for Multiobjective Flexible Job Shop Scheduling Problem with Total Energy Consumption Threshold[J]. IEEE Transactions on Cybernetics (S2168-2267), 2018, 49(3): 1097-1109.
- [10] 王雷, 蔡劲草, 石鑫. 基于改进遗传算法的多目标柔性作业车间节能调度问题[J]. 南京理工大学学报(自然科学版), 2017, 41(4): 494-502.
- Wang Lei, Cai Jincan, Shi Xin. Multi-objective Flexible Job Shop Energy-saving Scheduling Problem Based on Improved Genetic Algorithm[J]. Journal of Nanjing University of Science and Technology(Natural Science Edition), 2017, 41(4): 494-502.
- [11] 张新, 李珂, 严大虎, 等. 改进入侵杂草算法求解柔性作业车间调度问题[J]. 系统仿真学报, 2018, 30(11): 4469-4476.
- Zhang Xin, Li Ke, Yan Dahu, et al. Improved Intrusion Weed Algorithm for Solving Flexible Job Shop Scheduling Problem[J]. Journal of System Simulation, 2018, 30(11): 4469-4476.
- [12] 吴秀丽, 张志强, 杜彦华, 等. 改进细菌觅食算法求解柔性作业车间调度问题[J]. 计算机集成制造系统, 2015, 21(5): 1262-1270.
- Wu Xiuli, Zhang Zhiqiang, Du Yanhua, et al. Improved Bacteria Foraging Optimization Algorithm for Flexible Job Shop Scheduling Problem[J]. Computer Integrated Manufacturing Systems, 2015, 21(5): 1262-1270.
- [13] 吴虎胜, 张凤鸣, 吴庐山. 一种新的群体智能算法—狼群算法[J]. 系统工程与电子技术, 2013, 35(11): 2430-2438.
- Wu Husheng, Zhang Fengming, Wu Lushan. New Swarm Intelligence Algorithm-Wolf Pack Algorithm[J]. Systems Engineering and Electronics, 2013, 35(11): 2430-2438.
- [14] 董亚科, 杜军, 李博, 等. 多选择背包问题离散狼群算法研究[J]. 传感器与微系统, 2015, 34(6): 21-23, 26.
- Dong Yake, Du Jun, Li Bo, et al. Research on Discrete Wolf Pack Algorithm of Multiple Choice Knapsack Problem[J]. Transducer and Microsystem Technologies, 2015, 34(6): 21-23, 26.
- [15] Wu H S, Zhang F M, Li H, et al. Discrete Wolf Pack Algorithm for Traveling Salesman Problem[J]. Control & Decision (S1001-0920), 2015, 30(10): 1861-1867.
- [16] 李斌. 基于改进狼群算法的森林灭火资源调度研究[D]. 长沙: 中南林业科技大学, 2018.
- Li Bin. Research on Forest Fire-fighting Resource Scheduling Based on Improved Wolves Algorithm[D]. Changsha: Central South University of Forestry and Technology, 2018.
- [17] Wang F, Tian Y, Wang X. A Discrete Wolf Pack Algorithm for Job Shop Scheduling Problem[C]// 2019 5th International Conference on Control, Automation and Robotics (ICCAR). Beijing: IEEE, 2019.
- [18] 谢锐强, 张惠珍. 求解柔性作业车间调度问题的两段式狼群算法[J]. 计算机工程与应用, 2021, 57(7): 251-256.
- Xie Ruiqiang, Zhang Huizhen. Two-Vector Wolf Pack Algorithm for Flexible Job Shop Scheduling Problem[J]. Computer Engineering and Applications, 2021, 57(7): 251-256.
- [19] 张超勇, 饶运清, 刘向军, 等. 基于POX交叉的遗传算法求解Job-Shop调度问题[J]. 中国机械工程, 2004(23): 83-87.
- Zhang Chaoyong, Rao Yunqing, Liu Xiangjun, et al. An Improved Genetic Algorithm for the Job-Shop Scheduling Problem[J]. China Mechanical Engineering, 2004(23): 83-87.
- [20] 解潇晗, 朱晓春, 周琦, 等. 低能耗柔性作业车间调度研究[J]. 机电工程, 2020, 37(2): 132-137.
- Xie Xiaohan, Zhu Xiaochun, Zhou Qi, et al. Scheduling of Low Energy Consumption Flexible Job Shop[J]. Mechanical & Electrical Engineering Magazine, 2020, 37(2): 132-137.