赵子恒, 杨敬辉. 基于改进遗传算法的多样本柔性车间调度问题研究[J]. 智能计算机与应用,2026,16(5):204-211. DOI: 10.20169/j.issn.2095-2163.24071602

基于改进遗传算法的多样本柔性车间调度问题研究

赵子恒, 杨敬辉
(上海第二工业大学 计算机与信息工程学院, 上海 201209)

摘 要: 在工业 4.0 背景下,制造业正在经历向智能化与数字化的深刻转型,以提升效率并应对经济与环境双重挑战。本文针对这一趋势,专注于柔性车间调度问题(FJSP),以减少最大完工时间为目标,设计一种改进的遗传算法,采用双向量编码,由于 FJSP 是一个离散组合优化问题,本文的编码采用连续组合优化方式(DLGA)并通过层级竞争策略初始化种群,以增强种群多样性并促进算法跳出局部最优。算法设计中,种群被分为优胜者、跟随者和失败者 3 个层级,通过筛选、学习和竞争算子来优化解的质量,进一步提升了解决问题的能力。实验部分采用了两种类型的数据样本,一种是 Brandimarte 设计的 10 个多样化案例,进行多次实验。另一种为了保证算法的可用性,从实际保温杯工厂中进一步提取样本数据,证明算法的有效性。实验结果表明 DLGA 算法虽然在公共数据集上可以得到良好的效果,但从实际企业需求上来看,DLGA 可以更好地解决企业所面临的实际问题。

关键词: 工业 4.0; 柔性车间调度问题(FJSP); 改进遗传算法; 双向量编码; 层级竞争策略
中图分类号: 文献标志码: A 文章编号: 2095-2163(2026)05-0204-08

Research on multi-sample flexible job shop scheduling problem based on improved genetic algorithm

ZHAO Ziheng, YANG Jinghui
(School of Computer and Information Engineering, Shanghai Polytechnic University, Shanghai 201209, China)

Abstract: In the context of Industry 4.0, the manufacturing industry is undergoing a profound transformation to intelligence and digital transformation in order to improve efficiency and meet the dual challenges of economy and environment. Aiming at this trend, this paper focuses on the Flexible Job-Shop Scheduling Problem (FJSP), with the objective of minimizing the maximum makespan, and designs an improved genetic algorithm using double-vector encoding. Since FJSP is a discrete combinatorial optimization problem, this paper adopts a continuous combinational optimization method(DLGA) for encoding and initializes the population through a hierarchical competition strategy, which helps enhance population diversity and enables the algorithm to escape local optima. In the algorithm design, the population is divided into three levels: winners, followers and losers, and the quality of the solution is optimized by filtering, learning and competition operators. Which further enhances the ability to solve problems. Two types of data samples were used in the experiment. One was 10 diverse cases designed by Brandimarte, and multiple experiments were conducted. In order to ensure the availability of the algorithm, further sample data is extracted from the actual thermos cup factory to prove the effectiveness of the algorithm. The experimental results show that although DLGA algorithm can get good results on public data sets, DLGA can better solve the practical problems faced by enterprises in terms of actual enterprise needs.

Key words: Industry 4.0; Flexible Job Shop Scheduling Problem (FJSP); improved genetic algorithm; double-vector encoding; hierarchical competition strategy

0 引 言

在工业 4.0 的浪潮下,传统制造业正经历着一场深刻的变革,并开始迈向智能化与数字化的新时代。这场革命融合了人工智能与机器学习等技术,旨在模拟并优化真实的生产场景,推动生产车间从

作者简介: 赵子恒(1998—),男,硕士研究生。主要研究方向:柔性车间调度,企业资源规划,车间排程管理等。
通信作者: 杨敬辉(1968—),女,博士,教授级高级工程师。主要研究方向:企业系统集成,智能制造系统,调度与优化等。Email:jhyang@sspu.edu.cn。

收稿日期: 2024-07-16

传统模式向高度灵活，自动化的未来工厂转型。这一转型不仅提升了生产效率，也对企业应对经济与环境的双重重压下提出了新策略。面对经济增长与环境保护的双重挑战，制造业力求在可持续发展的框架下减少能耗，这已经成为其战略规划的核心。

在过去的半个多世纪，针对排程问题开展了诸多的研究工作，从上个世纪 60 年代开始，排程问题便引起了运筹学、工程技术以及数学领域科学家的高度关注，各种启发式算法被广泛应用于求解具有挑战性的 FJSP，由于该问题具有计算复杂性，实际应用中，多数的排程问题使用运筹学或启发式算法来求解，Pelges 等[9]成功将约束理论应用于重型卡车照明系统工厂，识别出制造过程中的瓶颈约束，运筹学方法自诞生以来在解决决策优化问题中存在优越性，尤其在优化生产排程问题上取得巨大成功。Hao 等[11]基于某离散制造企业的调度数据建立了多目标优化模型，提出了一种改进的局部搜索遗传算法(GALS)优化了调度模型的局部性能。

综上所述，本文以最小化最大总完工时间为目标，基于遗传算法对柔性生产车间调度排程算法展开研究。首先建立了混合流水车间环境下的数学模型。进而设计改进式的遗传算法对问题进行求解。在遗传算法中，本文采用双向量编码方式，采用连续组合优化方法，之后针对初始化种群采用层级竞争策略来寻找最优解。最后通过数据实验对算法性能进行验证。

1　问题的数学模型

1.1　问题描述

柔性车间调度问题(FJSP)涉及到多任务工件在多台可选加工设备上的调度安排，其中，每项工件的加工必须遵循预设的工序顺序，并且至少有一道工序可由多个可加工的机器执行，目标是在满足特定优化指标的前提下(如最小化生产周期或降低成本)，来制订最优的加工计划[15]。并且根据实际生产情况，做出以下合理假设：

(1)每个工件一次被一台机器加工；

(2)每台机器只可以一次处理一道工序；

(3)机器准备时间与工件顺序无关，且包含在处理时间中；

(4)每台机器的加工能力固定；

(5)暂不考虑物料在设备之间的运转时间；

(6)工件在某台机器上开始处理后，不允许中断，直至该工序完成；

(7)相邻工序之间的缓冲区容量无限。

1.2　目标与约束条件

1.2.1　参数模型

为了完整描述排程模型，需要给出参数定义，表 1 为金工车间排程模型所用参数表。

表 1　模型参数表

Table 1　Model parameters table

参数	描述
$C_{\max}$	最大完工时间
$S_{m,j}$	工件 j 在阶段 m 的开始时间
$t_{m,j}$	工件 j 在阶段 m 的加工时间
J	工件数
$x_{m,j,l}$	若工件 j 在阶段 m 的机器 l 上加工则为 1，否则为 0
L_m	工件在 m 阶段的结束时间
M	阶段总数

1.2.2 目标函数

在求解 FJSP 过程中，需要有目标函数来衡量优略程度，本文以优化最大完工时间为目标，即对应的目标函数为：

$$F = \min(C_{\max}) = \min(\max(s_{m,j} + j_{m,j})) \quad (1)$$

1.2.3　约束条件

工件可以跳过不必要的工序，且每个工序只能在一台机器上进行加工，公式如下：

$$\sum_{l=1}^{L_m} x_{m,j,l} \leqslant 1, \forall j \in \{1,2,\cdots,J\}, m \in \{1,2,\cdots,M\} \quad (2)$$

工件的各个工序需严格按照先后顺序进行加工，公式如下：

$$s_{m+1,j} - s_{m,j} \geqslant 0, \forall m \in \{1,2,\cdots,M-1\} \quad (3)$$

工件各工序的完工时间由上一工序的完工时间和当前工序的加工时间决定，公式如下：

$$s_{m+1,j} - s_{m,j} \geqslant t_{m,j}, \forall m+1 \in \{1,2,\cdots,M\} \quad (4)$$

2　遗传算法的设计

2.1　编码与解码

本文采用双向量编码。以表 2 为例，在此编码体系中，J_i 用来标记第 i 个工件，$O_{i,m}$ 表示工件 J_i 需要完成的第 m 个加工步骤，而 M_i 表示用于执行加工的第 i 台机器。一共 4 个工件，这些工件涉及 8 道总工序并且可利用 4 台不同的机器来完成加工任务，表格数据具体展现了各项工序在各台机器上的预计加工时长，空白则明确表示某工序无法在对应的机器上执行。

表 2 待加工工件实例表

Table 2 Examples of workpieces to be processed

工件	工序	机器编号			
		M1	M2	M3	M4
J_1	$O_{1,1}$	3			
	$O_{1,2}$		4	10	
J_2	$O_{2,1}$	5			
	$O_{2,2}$		5	10	
	$O_{2,3}$				10
J_3	$O_{3,1}$	5			
	$O_{3,2}$		5	2	
	$O_{3,3}$				5

2.1.1 编码

在柔性生产环境的调度框架下,问题被分解为两个任务,机器选择和工序选择。为了应对这样的问题,采用了分段编码策略。将染色体结构划分为工序编码区与机器编码区,如图 1 所示,工序编码区从左至右,工件编号第一次出现就代表该工件的第一道工序,以此类推;机器编码区:通过一些列数字明确指示每道工序应选用的加工设备。如图 1 所示,J_1 的第一道工序 $O_{1,1}$ 在机器 $M1$ 上加工,J_1 的第二道工序 $O_{1,2}$ 在机器 $M2$ 上加工;J_2 一共要经过 3 道工序的加工,所以机器编码的后 3 位数字 1,3,4 代表 $O_{2,1}$ 在 $M1$ 上加工,$O_{2,2}$ 在 $M3$ 上加工。

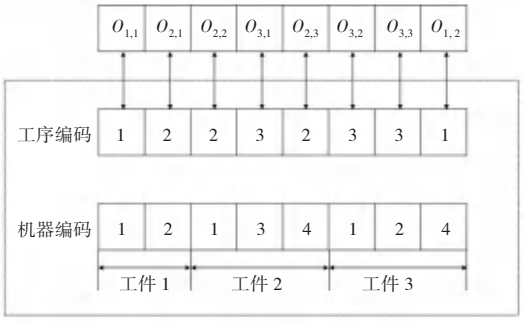


图 1 染色体结构示意图

Fig. 1 Schematic diagram of chromosome structure

2.1.2 解码

解码过程是指将已编码的信息逐步转化为具体的调度指令。此过程中,系统会按照工序编码的顺序,从左至右逐一解析每个代码的意义,将其转换为相应的操作命令。通过这样的解析帮助管理人员清晰了解资源分配和任务进展,如图 2 所示。

FJSP 属于离散组合优化问题,因此种群在编码之前还需要建立一个过渡机制,本文采用 Yuan 等[15]提出的过渡方法。为了实现离散优化到连续优化问题的转换,机器选择的过渡过程如下式:

$$y(i) = \left\{ \frac{2n}{(r(i) - 1)} \right\} (g(i) - 1) - n, r(i) \neq 1 \tag{5}$$

其中,$y(i)$ 表示编码的第 i 个数据;$r(i)$ 表示第 i 个元素可以选择进行相应操作的机器的数量;$g(i)$ 表示所选机器的序列号;如果 $r(i) = 1$,则 $y(i)$ 可取区间$[-n, n]$ 内的任意值。

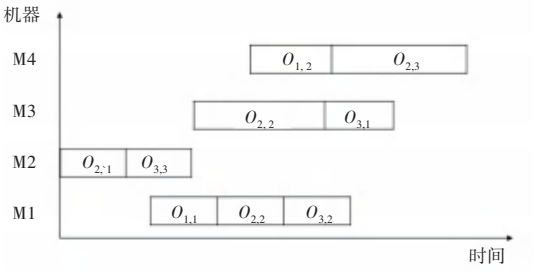


图 2 解码甘特图

Fig. 2 Decoding Gantt chart

操作序列的编码流程如下:首先编码在$[-n, n]$ 之间随机产生实数,随后为每个生成的实数按升序分配一个唯一的 ROV 数据点,因此,每个 ROV 数据点对应一个工序操作。然后根据作业的顺序对数据点进行重新排序,并根据重新排序后的数据点对实数进行重新排序,具体过程如图 3 所示。

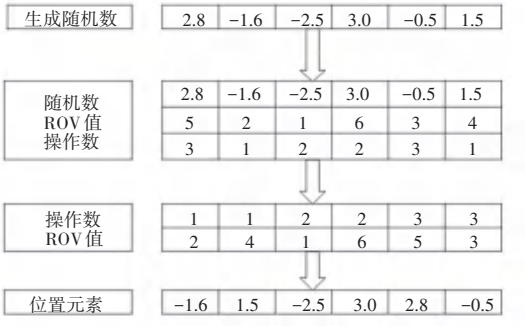


图 3 操作序列映射操作

Fig. 3 Sequence of operations map operations

解码流程与编码流程相反,首先是机器选择部分,根据公式(6)的逆运算推导得到机器选择的解码公式:

$$g(i) = \text{round}\left\{ \frac{[y(i) + n][r(i) - 1]}{2n} + 1 \right\} \tag{6}$$

操作顺序部分:本文按升序为每一个元素分配 ROV 数据点。然后,将 ROV 数据点作为固定位置数。最后通过将 ROV 数据点与作业进行匹配,实现作业置换。转换过程如图 4 所示。

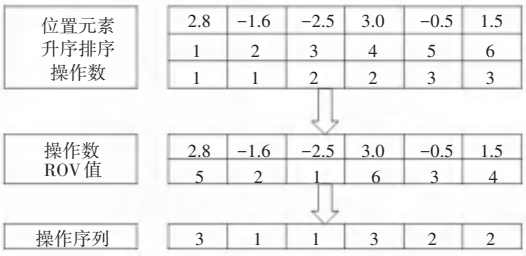


图 4　操作序列映射逆操作

Fig. 4　Sequence of operations maps inverse operations

2.2　基于社会竞争的遗传算法设计

(1)初始化

本文以混合方式使用一些不同的策略来生成具有初始解的种群。以局部搜索、全局搜索和随机搜索相结合的方式产生初始化种群,从而保证了初始解的质量与种群的基因多样性,大幅降低了获得可行解所需要的时间。

(2)基于层级竞争策略

将种群初始化之后根据适应度函数进行分层,共分为 3 个级别:优胜者仅使用筛选算子进行筛选,保留更优的解,维持恒定的数目,将淘汰的解驱逐至跟随者。跟随者随机选取优胜者进行跟随,使用学习算子,学习好的基因序列,改进自身基因。跟随者内部互相竞争,采用轮盘赌的方式保持恒定数目,将淘汰的解驱逐至失败者序列。失败者序列可随机抽取跟随者的基因片段或自身发生基因突变,防止自身被淘汰。整个种群数目恒定,设置 20% 淘汰率,从失败者、跟随者中进行淘汰,并随机产生新解扩充种群多样性。框架如图 5 所示。

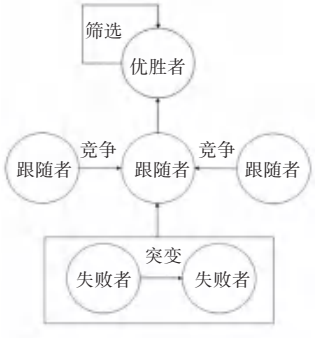


图 5　基于层级竞争序列框架

Fig. 5　Hierarchical competition sequence framework

离散解和连续解协同进化,离散域和连续域的解均基于适应度进行社会分层,分为优胜、跟随、失败 3 层,各自占比 0.3、0.3、0.4,呈现金字塔结构,优胜者只筛选,跟随者向优胜者学习,并互相竞争,失败者向跟随者学习或突变。

(3)筛选

使用反射对称的思路,将解进行自我反射或随机突变,之后添加到优胜群里,进行基因筛选,并添加上下边界显示,lb 表示下界,ub 表示上界,仅保留 20% 的优胜者,其余淘汰。

$$x_{\text{new}} = r(x_i - \text{lb}) + (1 - r)(x_i - \text{ub}), r \in \text{rand}\{0,1\} \tag{7}$$

$$x_{\text{new}} = \text{rand}\{\text{lb}, \text{ub}\} \tag{8}$$

(4)学习

学习算子则是随机选择一位优胜者 X_i,将自身的基因向优秀者 X_i 靠近,更新公式如下:

$$x_{\text{new}} = r \times x_j + (1 - r) \times x_i + \delta, r \in \text{rand}\{0,1\}, \delta \in \text{rand}\{0, 0.1 \times (\text{ub} - \text{lb})\} \tag{9}$$

(5)竞争

对于竞争部分,采用交换基因的竞争方法。该方法的详细竞争过程如下:

步骤 1　在父代基因中随机选择两个对应的位置。

步骤 2　交换基因产生后代。

该竞争算子的操作过程如图 6 所示。

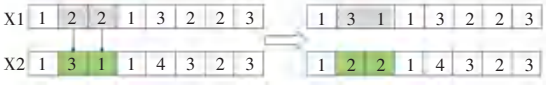


图 6　竞争算子

Fig. 6　Competition operator

(6)变异

引入变异是为了激发种群探索能力,本文所提到的工序编码的基因变异概率均被赋予 20% 的概率阈值,选定位置点出现变异则产生新的值。这样又会避免过渡变异可能引起的基因质量下降。

3　实验结果

3.1　仿真实验

本研究的实验环境基于 Core i5 处理器,目的是为了检验算法效能,评估方法采纳了 Brandimarte[22] 针对刹车盘问题所设计的 10 种多样化案例,这些案例涵盖了 10 到 20 个工件以及 6 到 15 台机器的不同配置。在对比分析中本文注意到张腾飞等[14] 之前的工作与本研究都聚焦于最小化最大完工时间。为了确保对比的公正性与一致性,本文在实验设计中沿用了张腾飞等[14] 算法的原始参数设置本文所提出的改进遗传算法调整了关键参数,具体包括设置种群大小为 100,迭代次数为 200,并实施了与张腾飞等[14] 算法、本文算法(LGA),以及离散连续组

合优化(DLGA)的混合方法,进行了全面的性能测试。每组数据均重复运行 10 次,以增加结果的可靠性和稳定性。最终的实验成果汇总于表 3 中,其中 n 代表工件数量,m 代表机器数量,两者相乘($n \times m$)定义了问题的规模,表 3 中分别列出了每组实验的最佳值(best)、最差值(worst)及平均值(avg),并以加粗字体突显出每次实验中的最优表现,以此直观展示算法改进带来的性能提升与鲁棒性优势。

对比实验数据分析揭示,在当前的样本规模下,不论是取得的最佳解、最差解还是平均解,组合优化的方式都优于或至少等同于其他两种算法,为了直观展示算法的高效性和调度逻辑的清晰性,特别选取了一个典型规格(10 工件和 6 台机器)的柔性车间调度实例,并以甘特图的形式予以呈现。如图 7 所示,通过采用不同颜色的区块来区分不同的工件,同一颜色区块沿横轴的顺序排列展现了工件的工序流程,区块的长度反应了相应工序的加工持续时间,而每个色块上的标签 i 明确指示了工件的编号。

表 3 三种算法仿真算例上的结果(t/min)

Table 3 Results of simulation examples of the three algorithms(t/min)

编号	$m \times n$	DLGA			LGA			GA		
		best	worst	avg	best	worst	avg	best	worst	avg
1	10×6	**146**	**168**	**160**	153	170	162	152	170	164
2	10×6	**149**	167	**156**	154	**160**	158	151	169	163
3	15×8	**195**	**206**	**200**	202	221	211	200	216	206
4	15×8	**197**	**211**	**206**	202	221	210	204	218	212
5	10×8	**114**	**123**	**118**	115	124	120	117	124	122
6	10×15	**143**	**146**	**145**	147	157	149	144	**146**	**145**
7	20×5	**366**	**388**	**374**	376	407	390	**366**	402	386
8	20×10	**219**	**232**	**233**	245	262	250	230	260	240
9	20×10	**220**	**245**	**233**	232	251	238	227	265	244
10	20×15	**176**	**195**	**183**	203	197	**183**	187	203	197

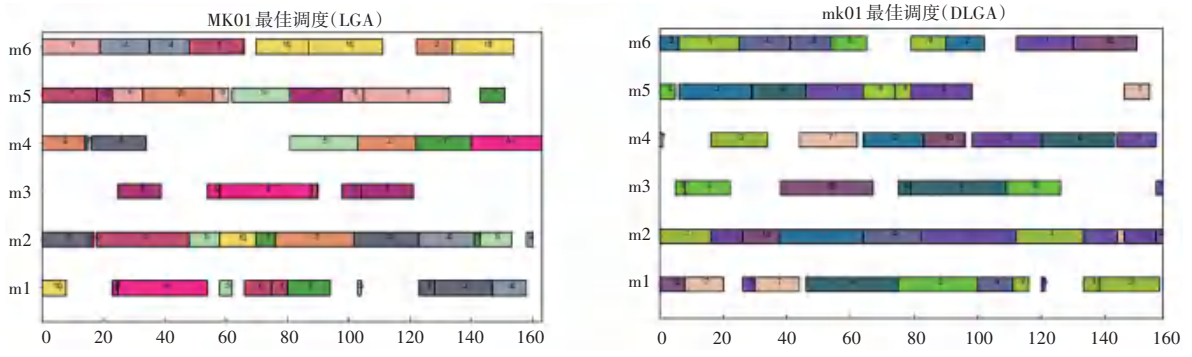


图 7 测试用例 10×6 的最优调度甘特图

Fig . 7 Optimal scheduling Gantt chart for test case 10×6

3.2 相关数据整理

为了紧贴实际情况,以 SL 工厂为例,本文选择了以下实际数据:产品的名称、数量,还明确了其加工流程链(见表 4)。值得注意的是,保温杯的半成品加工流程包括烘洗、抛口、过绵纶、砂管缝、内胆吹灰、擦底及检验共 7 个步骤,但检验作为最终的质量把关,不直接参与排产计划。在不同产品中,流程链 R1 包含了全部工序,而 R2 则根据产品特性省略了砂管缝这一环节。

表 4 待排程工单表

Table 4 Work order to be scheduled

产品名称	数量	流程链
红酒杯	100	R1
1.0 手柄子弹头	100	R2
500 mL 可乐瓶	110	R2
1.2 风尚旅游壶	100	R2
悠上运动瓶	100	R1

在金工车间实际生产的流程中,当产品加工数量较大时,加工物料并非一个一个转运,而是成批次的,通常将 100 单位数量的物料作为一批。在前文

算法的编码设计中，工件是编码的最小单位，为了减少编码的长度，加快算法的运行效率，单位数量 100 为一个虚拟工件，将如上几个工单拆分为如表 5 所示的几个代加工工件。

表 5　代加工工件表

Table 5　Workpieces to be machined

工件序号	工单编号	产品名称	流程链
1	SL-VB011	红酒杯	R1
2	SL-VB011	红酒杯	R1
3	SL-VB011	红酒杯	R1
4	SL-100FZ1	1.2 风尚旅游壶	R2
5	SL-100FZ1	1.2 风尚旅游壶	R2
6	SL-100FZ1	1.2 风尚旅游壶	R2
7	SL-100FZ1	1.2 风尚旅游壶	R2
8	SL-100FZ1	1.2 风尚旅游壶	R2
9	SL-100YD	悠上运动瓶	R2
10	SL-100L3	手柄子弹头	R2
11	SL-35ZL	可乐瓶	R2
12	SL-35ZL	可乐瓶	R2
13	SL-35ZL	可乐瓶	R2
14	SL-35ZL	可乐瓶	R2
15	SL-35ZL	可乐瓶	R2

抛光车间负责的是所有保温杯的抛光流程，成型车间生产的半成品进入抛光车间后，需依次经过烘洗无尾底，抛口，砂管缝，杯身磨砂，过绵纶，内胆吹灰，检验等工序，该车间的设备以半自动化设备为主，物料的转运主要依赖人工操作，运送到车间的半成品种类繁多。因此计划排程与管理难度较大，是高级计划与排程系统的重点研究对象。为清晰展示抛光车间的工序耗时与资源分配情况，现将可选用产线信息汇总为表 6。为保证实验对比的公平性，本文延用张腾飞等[14]设计算法中的原有参数，改进算法参数设置值：种群规模为 100，迭代次数为 400。实验环境为 Win10 系统，Python3.8，所有算法针对案例数据运行 10 次，最终算法的结果见表 7。将表 7 数值数据以折线图形式可视化呈现，得到图 8。从图 8 中可以清晰观察蚕桑：DLGA 的折线走势最为平稳，体现出较强的鲁棒性；LGA 的折线整体处于较低区间，在第 6 次实验取得最优值，但后期存在一定波动；GA 的折线波动幅度最大，稳定性最弱。

表 6　可选用产线表

Table 6　Optional production line table

产线序号	产线名称	对应工序
M1	烘洗产线 1	烘洗产线
M2	烘洗产线 2	烘洗产线
M3	烘洗产线 3	烘洗产线
M4	擦底 1	擦底
M5	擦底 2	擦底
M6	过绵纶 1	过绵纶
M7	过绵纶 2	过绵纶
M8	抛口 1	抛口
M9	抛口 2	抛口
M10	抛口 3	抛口
M11	抛口 4	抛口
M12	砂管缝 1	砂管缝
M13	砂管缝 2	砂管缝
M14	砂管缝 3	砂管缝
M15	砂管缝 4	砂管缝
M16	内胆吹灰 1	内胆吹灰
M17	内胆吹灰 2	内胆吹灰
M18	内胆吹灰 3	内胆吹灰
M19	内胆吹灰 4	内胆吹灰

表 7 三种算法十次实验的结果(t/min)

Table 7 Results of ten experiments of three algorithms(t/min)

算法	实验次数									
	1	2	3	4	5	6	7	8	9	10
DLGA	1 815	**1 791**	1 871	1 868	1 819	1 826	1 872	1 851	1 848	**1 811**
LGA	**1 792**	1 801	**1 798**	**1 775**	**1 777**	**1 755**	1 842	**1 782**	**1 771**	1 815
GA	1 810	1 838	1 805	1 789	1 804	1 831	**1 823**	1 826	1 815	1 848

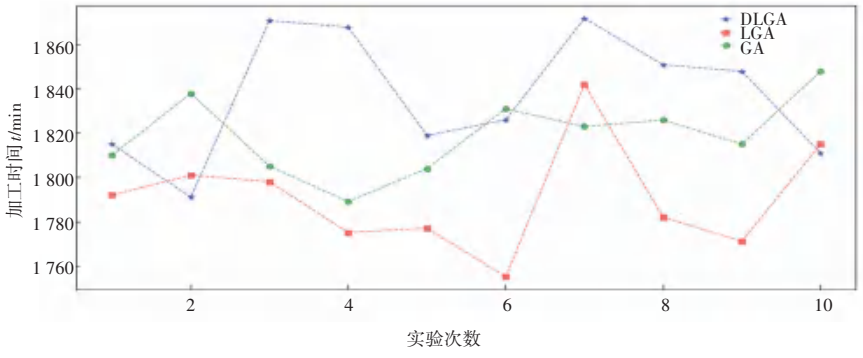


图 8 三种算法 10 次结果折线图

Fig. 8 Line graph of 10 experimental results of the three algorithm

4 结束语

本文针对工业 4.0 背景下的制造业智能化与数字化转型需求,深入探讨了柔性车间调度问题(FJSP)。提出一种改进的遗传算法,采用双向量编码方式与层级竞争策略,结合学习算子和竞争算子来提高阶段多样性和算法的探索能力。实验结果表明,改进遗传算法 LGA 在处理 Brandimate 设计的数据时表现优异。但从实际企业的数据规模来看 DLGA 的处理能力优于 LGA 算法,为将来企业实现生产效率和资源优化配置提供了有力支持。

参考文献

[1] GEEM Z W, KIM J H, LONGANATHAN G V. A new heuristic optimization algorithm: Harmony search[J]. Simulation, 2001, 76(2): 60-68.

[2] LEE K S, GEEM Z W. A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice [J]. Computer Methods in Applied Mechanics and Engineering, 2005, 194(36/37/38): 3902-3933.

[3] LEE K S, GEEM Z W, LEE S, et al. The harmony search heuristic algorithm for discrete structural optimization [J]. Engineering Optimization, 2005, 37(7): 663-684.

[4] MAHDAVI M, FESANGHARY M, DAMANGIR E. An improved harmony search algorithm for solving optimization problems[J]. Applied Mathematics and Computation, 2007, 188(2): 1567-1579.

[5] GEEM Z W, LEE K S, PARK Y. Application of harmony search to vehicle routing[J]. American Journal of Applied Sciences, 2005, 2(12): 1552-1557.

[6] VASEBI A, FESANGHARY M, BATHAEE S M T. Combined heat and power economic dispatch by harmony search algorithm [J]. International Journal of Electrical Power & Energy Systems, 2007, 29(10): 713-719.

[7] WANG L, PAN Q K, TASGETIREN M F. Minimizing the total flow time in a flow shop with blocking by using hybrid harmony search algorithms[J]. Expert Systems with Applications, 2010, 37(12): 7929-7936.

[8] DEL SER J, MATINMIKOO M, GIL - LOPEZ S, et al. Centralized and distributed spectrum channel assignment in cognitive wireless networks: A harmony search approach [J]. Applied Soft Computing, 2012, 12(2): 921-930.

[9] PEGELS C C, WATROUS C. Application of the theory of constraints to a bottleneck operation in a manufacturing plant[J]. Journal of Manufacturing Technology Management, 2013, 16(3): 302-311.

[10] ABU - SHARMS M, RAMADAN S, Al - DAHIDI, et al. Scheduling large-size identical parallel machines with single server using a novel heuristic - guided genetic algorithm (das/ga) approach[J]. Processes, 2022, 10(10): 2071.

[11] WANG H, WANG Y, LV X, et al. Genetic algorithm with local search for the multi-target scheduling in flexible manufacturing system[J]. Journal of Circuits, Systems and Computers, 2022, 31(16): 2250279.

[12] WANG Cuiyu, LI Yang, LI Xinyu. Solving flexible job shop scheduling problem by a multi - swarm collaborative genetic algorithm[J]. Journal of Systems Engineering and Electronics, 2021, 32(2): 261-271.

[13] SANG Y, TAN J. Intelligent factory many-objective distributed flexible job shop collaborative scheduling method[J]. Computers & Industrial Engineering, 2022, 164: 107884.

[14] 张腾飞, 马跃, 李力, 等. 柔性作业车间调度问题的改进遗传算法[J]. 小型微型计算机系统, 2017, 38(1):129-132.

[15] YUAN Y, XU H, YANG J. A hybrid harmony search algorithm for the flexible job shop scheduling problem[J]. Applied Soft Computing, 2013, 13(7): 3259-3272.
[16] 王子文. 电站制造业高级计划与排程系统的设计实现[D]. 上海:上海交通大学,2013.
[17] 李聪波, 寇阳, 雷焱绯, 等. 基于动态事件的柔性作业车间重调度节能优化[J]. 计算机集成制造系统, 2020, 26(2): 288-299.
[18] YI W, YUAN Y, HOSEINNEZHAD R, et al. Resource scheduling for distributed multi-target tracking in netted colocated MIMO radar systems[J]. IEEE Transactions on Signal Processing, 2020, 68(2): 1602-1617. DOI:10.1109/TSP.2019.2956644.
[19] LI Z, XIE J, ZHANG H, et al. Adaptive sensor scheduling and resource allocation in netted collocated MIMO radar system for multi-target tracking[J]. IEEE Access, 2020, 8: 109976-109988.
[20] ZHANG P, CHEN N, XU G, et al. Multi-target-aware dynamic resource scheduling for cloud-fog-edge multi-tier computing network[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(5):3885-3897.
[21] 顾笑言. 基于群体智能优化算法的车间调度关键技术研究与开发[D]. 济南:齐鲁工业大学,2024. DOI:10.27278/d.cnki.gsdqc.2024.000069.