

doi: 10.12052/gdutxb.210001

改进遗传算法求解基于MPN混流制造车间调度问题

王美林, 曾俊杰, 成克强, 陈晓航

(广东工业大学 信息工程学院, 广东 广州 510006)

摘要: 大规模混流制造系统存在规模大、资源约束多的特点, 造成在作业调度时产生维数灾难, 从而产生搜索求解难的问题。本文针对此类问题, 在基于(Manufacturing Petri Net, MPN)模型的基础上, 提出一种改进遗传算法进行求解。首先, 重新定义了染色体的结构, 并采用染色体安排段压缩求解的搜索空间。其次, 在染色体交叉环节用粒子群优化(Particle Swarm Optimization, PSO)优化机制引导染色体优化方向, 在染色体变异环节用模拟退火算法(Simulated Annealing Algorithm, SAA)的机制防止遗传算法的过早收敛。然后, 在每一次种群迭代后对当前最优个体采用邻域搜索机制尝试拔高最优个体的适应度。实验数据表明, 改进遗传算法在求解的最优性方面有了较大改进。

关键词: 遗传算法; 混合流水车间; 最短完工时间

中图分类号: F224.32

文献标志码: A

文章编号: 1007-7162(2021)05-0024-09

A Research on Improved Genetic Algorithm for Flexible Job Shop Scheduling Problem Based on MPN

Wang Mei-lin, Zeng Jun-jie, Cheng Ke-qiang, Chen Xiao-hang

(School of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: The large-scale HFS (Hybrid Flow Shop) system has the characteristics of large scale and many resource constraints, which causes dimension disasters during job scheduling, which results in difficult search and solution problems. Addressing such problems, an improved genetic algorithm is proposed, based on the MPN (Manufacturing Petri Net) model, with the makespan as the optimization goal. Firstly, the structure of the chromosome is redefined, and the corresponding algorithm of chromosome arrangement segment used to compress the search space. Secondly, the particle swarm optimization (PSO) optimization mechanism is used to guide the direction of chromosome optimization in the chromosome crossover operator, and the simulated annealing algorithm (SAA) mechanism used in the chromosome mutation operator to prevent the premature convergence of the genetic algorithm, and then after each population iteration, the current maximum individual uses the neighborhood search mechanism to try to improve the fitness of the best individual. Finally, the experimental data shows that the improved genetic algorithm has made great improvements in the optimization of the solution.

Key words: genetic algorithm; hybrid flow shop; makespan

混流制造(Hybrid Flow Shop, HFS)是一种常用且主流的大批量定制生产(Mass Customization Production, MCP)组织模式, 可以通过定制阶段并行机上的工艺参数满足客户多样化的产品需求。HFS本质上是一个NP-Hard难题^[1], 而其系统规模大、资源约束多的特性, 导致了解空间过大难以求解的

问题。

国内外很多研究团队对混流制造的调度问题进行了深入的研究。何军红等^[2]针对在MES系统(Manufacturing Execution System, MES)生产调度模块的柔性作业车间问题, 提出了3个阶段的优化调度算法, 对具有10个库所、6个工件的HFS问题进行求解。王

收稿日期: 2021-01-05

基金项目: 国家自然科学基金资助项目(U1701266); 广东省知识产权与大数据重点实验室项目(2018B030322016); 广东省科技计划项目(2019A050513011, 2017B090901056); 广州市科技计划项目(202002030386)

作者简介: 王美林(1975-), 男, 副教授, 主要研究方向为制造物联网和工业互联网使能技术、生成过程调度与优化等

通信作者: 曾俊杰(1996-), 男, 硕士研究生, 主要研究方向为动态生产调度、制造物联网等, E-mail: 1277742051@qq.com

秀萍^[3]针对柔性车间提出一种具有优先级的启发式算法,能够对10个库所、10个工件的问题求解。而何涛^[4]针对智能制造柔性车间采用了改进过的遗传算法,将机器的选择与工件的顺序进行分段,对6个库所、4个工件的加工问题进行求解。在求解调度问题中,研究人员尝试了各类智能算法。如黄亚平^[5]在基于汽轮机的智能调度系统上采用改进的蚁群算法解决生产调度问题,在调度任务发生变化时快速地根据上次调度结果进行重调度。又如田松龄^[6]针对柔性车间,采用异步仿真时钟的蚁群并行搜索算法,减低资源共享型系统的运算成本。白俊峰^[7]针对车间调度问题采用包含了精英保留策略与逆转变异机制的遗传算法,其在求解7个工件、5台机器的案例中能够有较好的效果。余璇^[8]针对柔性调度问题提出一种混合遗传禁忌搜索算法,对问题模型进行优化并能够在 10×10 的案例中获得不错的效果。吴大立和田海霖等^[9-10]针对Petri网建模的车间作业调度问题,采用数学的方法对染色体编码并提出基于GA(Genetic Algorithm)算法的优化求解。但是以上算法多是针对特定场景较小规模问题进行求解,在面临大规模问题时都无法避免地形成巨大的解空间或陷入局部解。

本文针对HFS大规模HFS系统的调度难题,使用MPN(Manufacturing Petri Net)^[11]建模方式进行模型压缩,采用2段编码方式重塑染色体限定搜索范围,用包含粒子群机制(Particle Swarm Optimization, PSO)交叉子、模拟退火机制(Simulated Annealing Algorithm, SAA)变异子与邻域搜索机制的改进遗传算法来改进解搜索方向以优化大规模HFS问题的求解。

1 问题描述

混合流水车间问题可以描述为:一共有 n 个待加工工件,它们需要经过 s 道工序进行加工,每道工序会存在 M_i 台并行机器($M_i \geq 1; i = 1, 2, \dots, s$);同一工序上不同类型的机器加工同一工件的加工时间会有所不同;而且每个工件都要经过每一道工序进行加工;当工件被排产到第 i 道工序时可以被该工序中的 M_i 台并行机中的任意一台机器加工;而且车间中的所有机器都是非抢占式。

以图1为例,图1描述了一个包括了2个阶段工序的HFS系统。工序1有M型并行机2台和N型并行机2台,共4台,工序2有H型并行机2台和G型并行机2台一个共4台。对图1中的HFS系统采用MPN模型进行同类并行机压缩建模后,可以将一个 4×2 的HFS模型

压缩成一个 2×2 的HFS MPN模型,其中资源约束相同的同型设备被建模成MPN中的一个place节点。在MPN模型中,HFS作业的调度问题转换为在制品工件Token,在目标函数监督下,全部从库所 p_0 转移到 p_i 的路径搜索问题。

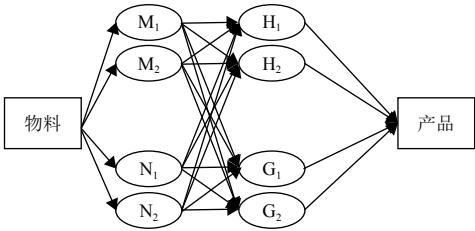


图 1 混合流水车间案例
Fig.1 Case of hybrid flow workshop

2 模型构建

2.1 模型建立

本文针对以上基于MPN的HFS调度,为了适应改进GA算法压缩解搜索空间的需求,在MPN的Petri模型基础上,加入用于描述工件加工可选路径的约束和生成作业集约束,避免算法运行时的盲目搜索。模型案例如图2所示。

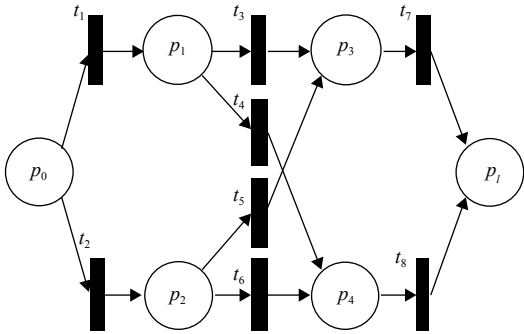


图 2 基于案例的MPN模型
Fig.2 Case-based MPN model

本文的MPN模型定义如下^[11]:

- (1) $P = \{p_1, p_2, \dots, p_l\}$ 是模型中库所的有限集合,表示生产车间中的同类型并行机集合。
- (2) $E = \{t_1, t_2, \dots, t_l\}$ 是有限的集合,用于生产状态变化的触发器。
- (3) I, O 分别表示工件离开库所和进入库所的路径。
- (4) S 是MPN模型中所有库所令牌的状态向量, $S_\tau(p_i)$ 表示库所 p_i 在时间 τ 时的所有令牌数量。 S_0 表示MPN的初始状态, S_l 表示MPN的结束状态。
- (5) $K(p_i)$ 是一个函数表示库所 p_i 内并行机的数量也就是库所的容量。

(6) $\Theta(\theta) = \{\Phi\}$ 是工件类型函数,能够计算得出工件 θ 的类型集合 Φ , Φ 表示工件类型的集合即 $\Phi \in \{c_1, c_2, \dots, c_\mu\}$ 。

(7) D 是各种不同类型的工件在库所中进行加工的时间矩阵。

(8) $W(M)$ 是MPN模型中从库所 p_0 到 p_i 的加工路径所组成的向量集合。 r 为路径向量的索引。 $W(M)_r$ 表示 $W(M)$ 中的第 r 条路径。如图2所示,其 $W(M) = \{<p_1, p_3>, <p_1, p_4>, <p_2, p_3>, <p_2, p_4>\}$ 。

(9) $Q_{(\theta, p_i)} = \{(\theta, p_i) | \theta \in \text{Order}, p_i \in P, (\theta, p_i) \in \theta \times W(M)_r\}$,是表示工件 θ 选择了路径 $W(M)_r$ 完成所有操作所产生的作业集合。如工件1选择了2号路径($W(M)_2$)则其作业集为 $\{Q_{(1, p_1)}, Q_{(1, p_4)}\}$ 。

(10) $s((\theta, p_i))$ 、 $c((\theta, p_i))$ 分别表示工件 θ 在库所 p_i 上的进入时间与离开时间。且 $s((\theta, p_i)) - c((\theta, p_i)) \geq D(\Theta(\theta), p_i)$ 。

(11) Order是进入MPN的一个加工订单,包含了一定数量不同类型工件token集合。

2.2 优化目标

$$\min C = \min(F(S_0 \rightarrow S_i)) \quad (1)$$

$$\text{s.t.} \begin{cases} \sum_{r=1}^R m_{\theta, r} = 1 \quad (\theta \text{选择} r \text{路径} x = 1 \text{否则} x = 0) & (2) \\ Q_{(\theta, r)} = m_{\theta, r} \times \theta \times W(M)_r & (3) \\ S_{\tau}(p_i) \leq K(p_i) & (4) \end{cases}$$

其中, $m_{\theta, r} \begin{cases} = 1 & \text{如果令牌}\theta\text{选中了}r\text{路径} \\ = 0 & \text{否则} \end{cases}$ 。

式(1)表示优化目标为时间最短, $F(S_0 \rightarrow S_i)$ 表示MPN从开始状态 S_0 (所有工件都在 p_0)一直到结束状态 S_i (所有工件进入 p_i)所使用的时间。约束(式(2))表示任何要加工的工件都要选择一个路径 r , R 为路径总数量。式(3)为优先约束,表示工件 θ 选择了路径 r 进行加工,会按照向量 $W(M)_r$ 的顺序经过向量中的库所。式(4)为资源约束,表示在MPN模型加工的任意时刻 τ 下,库所同时加工工件数量 $S_{\tau}(p_i)$ 不超过该库所的最大容量。

2.3 基于MPN模型的染色体编码

目前处理柔性加工问题的染色体结构有两种,将作业的选择与安排混合考虑^[12]或分开考虑^[13]。本文采用后者,将一个完整的染色体分为:路径选择段,作业安排段。选择段为染色体的第一段 genestring_0 用于描述约束(式(2))。其余部分为每个工序阶段对应一个安排段,描述每个阶段所有加工作业的优先约束(式(3))和资源约束(式(4))。

染色体的编码算法1:

Step1 编码定义优先约束:编码染色体的选择段。 genestring_0 确定每个工件 θ 的加工路径 r 。

Step2 根据优先约束编码定义每一个阶段的资源约束:

(1) 从第一个安排段 genestring_1 开始,令 $\text{var} = 1$ 。

(2) 把 $\text{genestring}_{\text{var}}$ 进行切割。 $\text{genestring}_{\text{var}}$ 按照工序 var 中的 place 数量切割,并把切割的份数赋值给变量 N 。

(3) 编排 $\text{genestring}_{\text{var}}$ 的 N 个子块,每个子块中 token 顺序代表对应 place 内作业加工顺序。

(4) $\text{var} \leftarrow \text{var} + 1$ 。如果 $\text{var} >$ 工序数,则End,否则goto (2)。

End

以图2的Petri网为例。假定 place 容量设置为 $\{K(p_i)\}^T = \{(2, 2, 2, 2) | (i \in [1, 4])\}$, Order由4个不同类型的工件组成,表示为 $\{\theta_1, \theta_2, \theta_3, \theta_4\}$ 、 Φ 分别为 $\{c_1, c_2, c_3, c_4\}$ 。图3为算法1生成的一个用染色体编码方案。 genestring_0 记录了Order中4个工件分别选择了1, 4, 4, 4号路径,形成了 p_1 作业的作业集合 $\{Q_{(1, p_1)}\}$ 、 p_2 作业的作业集合 $\{Q_{(2, p_2)}, Q_{(3, p_2)}, Q_{(4, p_2)}\}$ 、 p_3 作业的作业集合 $\{Q_{(1, p_3)}\}$ 、 p_4 作业的作业集合 $\{Q_{(2, p_4)}, Q_{(3, p_4)}, Q_{(4, p_4)}\}$ 。 genestring_1 记录了工序1库所 p_1 和 p_2 的工件处理顺序: p_1 只有工件1, p_2 工件处理顺序为2, 4, 3。

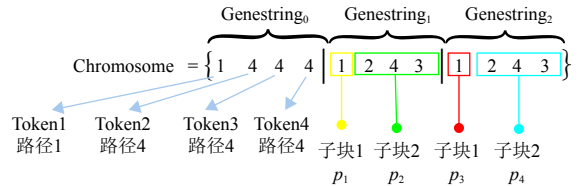


图3 染色体编码结果

Fig.3 Chromosome coding results

由染色体的编码算法1,可看出染色体将具体的调度方案分成了选择段与安排段。选择段确定了工件要以经过哪条加工路线的库所完成加工,即每个库所 p_i 会接收到什么作业,而安排段记录库所 p_i 内接收作业的处理顺序。当给出一个 D 矩阵时,图3中的染色体便能够获得该编码在MPN模型下所有作业的 $s((\theta, p_i))$ 、 $c((\theta, p_i))$,并可以将它们记录在一个 schedule 中。

3 染色体约束与优化

很多使用染色体对车间问题进行求解时都会采用调整算法去掉部分的无用解^[14-15],但由于调整方法简单且无法压缩解空间,所以对本文的大规模HFS问题不适用。在此本文设计了染色体安排段优化算法对染色体(解空间)的编码过程进行补充。

染色体安排段优化算法是在算法1编码的基础上,对 p_i 内接收作业(即其对应安排段)的处理顺序进行有限次数有反馈的调整。反馈机制是寻找拖慢调度的起始作业操作,并调整下次迭代中工序1的相应操作顺序来优化该操作。由于该算法通过反馈解决了调整方向的问题,而且该算法对相同选择段的染色体,执行一定次数迭代调整后,获得唯一最优的染色体安排段。

染色体安排段优化算法2:

Input:

Chromosome (算法1生成的一个初始基因),

DN (最大迭代次数)

Output:

<Chromosome, schedule> (优化后的染色体及其对应的时间报表)

Step1 染色体Chromosome的生成。保持工序1不变,其他工序的编码用算法1按照规则(1、到达时间最快,2、加工时间最短,3、工件编号最小)重新编排。

Step2 通过Chromosome输入MPN生成schedule并记录本次生成的染色体二元关系<Chromosome, schedule>。

Step3 取出记录中最优的二元关系<Chromosome, schedule>

(1) 从schedule取出每个工件的最后一个操作,将它们按完成时间进行降序排序,并令 $var \leftarrow 1$ 。

(2) 从排序结果中取出倒数第 var 个完成的作业 $Q(\theta_x, p_i)$ 。

(3) 根据作业 $Q(\theta_x, p_i)$ 更新 $Q(\theta_y, p_i)$ 和 $Q(\theta_x, p_j)$ ($Q(\theta_y, p_i)$ 代表同在库所 p_i 但顺序先于 $Q(\theta_x, p_i)$ 的作业; $Q(\theta_x, p_j)$ 代表工件 θ_x 的前一个工序作业)。

(4) 查找作业 $Q(\theta_x, p_j)$ 与 $Q(\theta_x, p_i)$ 之间是否存在空闲,若不存在空闲,则 $Q(\theta_x, p_i) \leftarrow Q(\theta_x, p_j)$;若存在空闲,则 $Q(\theta_x, p_i) \leftarrow Q(\theta_y, p_i)$ 然后go to Step3(3),直到 $Q(\theta_x, p_i)$ 的库所 p_i 在第一工序为止。

(5) 判断作业 $Q(\theta_x, p_i)$ 的加工库所 p_i 容量,如果小于或等于1即没有调整的余地go to Step3(6),否则 go to Step4表示找到调整位置进行调整

(6) $var \leftarrow var + 1$,如果 var 大于订单Order中的工件的数量,则End否则go to Step3(2)。

Step4 genestring₁中找到库所 p_i 的基因子段,并将 p_i 中的作业 $Q(\theta_x, p_i)$ 与它的前一个作业交换。go to Step5。

Step5 判断是否到迭代次数DN,如果是结束并输出<Chromosome, schedule table>,否则go to Step1。

End

4 算法设计

为了探索本文所研究的MPN模型下的HFS问题的优解,提出了一种改进的遗传算法(Improved Genetic Algorithm, IGA)。目前很多的遗传算法都会为其交叉变异添加新的机制^[16],但是无法处理大规

模问题。本算法结合了3种方法从3个方面解决了遗传算法在求解问题中所遇到的难题。(1) 本文会使用如图4的方式将染色体选择段与适应度进行关联,让遗传算法(IGA)把搜索的精力集中在搜寻最优的染色体选择段上,压缩可行解空间。(2) 本文的算法所使用的交叉子将采用粒子群优化(PSO)机制,而变异子将采用模拟退火算法(SAA)机制,这样的组合有利于改善遗传算法过早收敛的问题。(3) 算法还会对每次迭代的最优个体(Optimal Individual, OI)进行邻域搜索挖掘该个体的“潜力”。

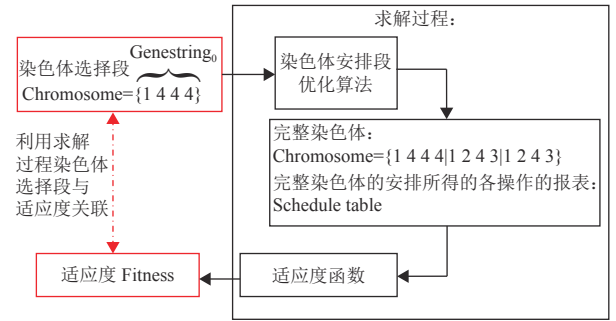


图4 以染色体选择段代替完整染色体与适应度进行关联

Fig.4 Using chromosome selection segments to connect with fitness

4.1 以染色体选择段代替完整染色体与适应度进行关联

交叉、变异、邻域搜索的运算只需要使用选择段即可,当需要进行染色体间比较时,染色体能够通过<Chromosome, schedule table>,获取详细信息并使用式(5)得出染色体的适应度来进行比较。这样便可在无需考虑无用解(不合理的染色体)的同时让遗传算法只去筛选选择段,压缩了解空间。之后出现的遗传算法的各种操作都是对染色体选择段进行操作,而且认为选择段就是一个完整的个体适应度函数。

$$\text{Fitness} = -C_{\max} \quad (5)$$

$C_{\max}$ 为schedule table中最大的值(最后一个作业的完成时间)。

4.2 粒子群算法与模拟退火法在遗传算法上的结合运用

4.2.1 交叉算子

为了提高算法收敛性,本文引入PSO机制(其中 GI_1 表示guided individual、 PI_1 表示 parent individual1、 PI_2 表示 parent individual2、 OI_1 表示 Offspring individual1、 OI_2 表示 Offspring individual2)。它们使用了特殊的交叉方式($GI_1 + PI_1 + PI_2 \rightarrow OI_1 \& OI_2$),从群内个体两两交叉生成两个子代,变为群外一个(GI_1)

与群内两个(PI_1 & PI_2)个体交叉获得两个子代,保证了后代个体 OI_1 、 OI_2 有明确的优化方向,使种群的收敛性提高。

然后为交叉引入 k_1 、 k_2 、 L 三个变量, k_1 表示从 GI_1 个体中学习的学習率, k_2 表示从另一个父代个体中学习的学習率, L 表示选择段的长度。具体交叉方式为(如图5):使用当前迭代所得的最优个体作为引导个体(GI_1)并随机抽出两个父代个体(PI_1 & PI_2)。后代 OI_1 以 PI_1 为基础(OI_2 以 PI_2 为基础),并学习 $(L/2) \times k_1$ 个 GI_1 的基因,学习方式直接将选中的基因覆盖在 PI_1 (PI_2)上如图5的黄色基因,然后再学习 $(L/2) \times k_2$ 个 PI_2 (PI_1)的基因,学习方式直接将选中的基因覆盖在 PI_1 (PI_2)上,但不可以选之前在 GI_1 中选中的位置,如图5的蓝色基因。

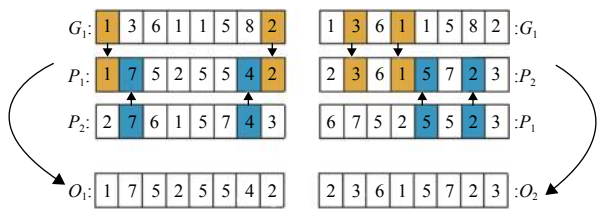


图5 交叉算子

Fig.5 Crossover operator

由于PSO机制的添加,引导个体(本群内最优染色体)参与群内交叉能够引导染色体的生产方向,提高收敛性。

4.2.2 变异算子

为了避免遗传算法过早收敛的弊端,本文改进了变异算子,引入新的模拟退火(SAA)机制,使用接收概率实现变异结果的条件接受,使用变异倾向函数实现变异的方向性指导,从而提高未出现基因序列的出现概率。两者组合可以帮助算法跳出局部最优解。

在本节中,变异算子设计如下:

Step1: 在个体中随机选择一个位置。

Step2: 在选择的位置上,通过突变趋势函数(式(6))和(式(7))所得出的可选基因元素(路径)的概率分布,从中按概率选取一个基因值(路径)替换原基因。

Step3: 通过模拟退火法的接受概率判断是否接受该变异所得的染色体。

Step4: 当代总群中的所有个体都执行过变异后,按照模拟退火中 $T_a = T_a \times K$ 的规则执行退火(T_a 为当前温度, K 为退火常数),改变温度,影响下一次算法迭代的接受率。

其中,突变趋势函数将对每次迭代后出现在个体上每个位置的值进行计数,并获得在每个位置突

变为不同值的相应概率,从而影响个体变异的倾向。

$$f_{mt}(\theta, G) = \left(\frac{1/\sum(r_1, \theta, G)}{\sum_{n=1}^N 1/\sum(r_n, \theta, G)}, \dots, \frac{1/\sum(r_N, \theta, G)}{\sum_{n=1}^N 1/\sum(r_n, \theta, G)}, \dots \right) \quad (6)$$

$$\text{sum}(r_n, \theta, G) = \sum_{g=G_0}^G f_s(r_n, \theta, g) \quad (7)$$

其中 $\theta \in \{\text{Token}\}$, $r_n \in \{\text{Path}\}$, $N \in |\text{Path}|$, G 为当前的种群代数, G_0 是开始的种群代数。 $f_s(r_n, \theta, g)$ 表示在第 g 代中的所有个体在某个位置(工件 θ)上选择了某个基因值(路径 r_n)的频率。

4.3 邻域搜索搜索最优个体

由于Petri网模型的拓扑结构的原因,Chromosome个体的细微差别,会导致适应度存在巨大差异。为了能够激发当前系统发现的最优个体(OI),在遗传算法执行过交叉与变异后都会进行最优个体邻域搜索。

邻域搜索算法:

Input:

OI_G (当前代最优个体),

OI_{G-1} (上一代最优个体, G 为当前代数)

Step1: 如果 OI_G 和 OI_{G-1} 不是同一个个体。goto step1(1), 否则goto step1(2)。

(1) 将 OI_G 输入到基本的邻域搜索函数得到新个体NI(NI: New Individual);

(2) 按照突变趋势函数(6)、(7)将 OI_G 上随机选择一个位置(工件 θ)有倾向的进行改变,然后将该个体输入到基本的邻域搜索函数得到新个体NI(New Individual);

Step2: 比较个体 OI_G 与个体NI的适应度,如果NI适应度大则 $OI_G \leftarrow NI$ 。

Output: OI_G

其中基本邻域搜索函数是一个输入为最优个体选择段,输出为其邻域(只有一个位置的值与genestring₀不同)内的最优染色体选择段。如果邻域内有多多个适应度相同的染色体则选择每个工件完成时间的总和最小的个体。

4.4 IGA算法框架

综合上述的算法模块,改进遗传算法的算法流程如图6所示。

5 实例分析

采用本团队所研究的基于覆铜板层压厂混流加工环境所构建的Petri网模型为基础^[1],并用同样的

表 3 田口实验
Table 3 Taguchi experiment

实验组编号	参数设置						结果		
	p_c	k_1	k_2	T_0	t_c	t_n	达到最优的代数	最优完成时间	种群内的平均完成时间
1	1	1	1	1	1	1	93.8	378.5	464.3
2	1	2	2	2	2	2	152.8	378.3	456.1
3	1	3	3	3	3	3	148.4	379.1	432.0
4	1	4	4	4	4	4	149.0	378.0	425.7
5	1	5	5	5	5	5	138.8	376.6	380.0
6	2	1	2	3	4	5	130.2	376.1	472.6
7	2	2	3	4	5	1	161.2	376.3	454.8
8	2	3	4	5	1	2	171.8	378.9	444.3
9	2	4	5	1	2	3	163.0	376.4	394.7
10	2	5	1	2	3	4	94.6	376.8	397.6
11	3	1	3	5	2	4	123.8	376.3	494.8
12	3	2	4	1	3	5	115.0	375.8	458.1
13	3	3	5	2	4	1	148.0	378.1	426.9
14	3	4	1	3	5	2	107.8	376.8	380.1
15	3	5	2	4	1	3	135.4	377.0	380.6
16	4	1	4	2	5	3	157.6	377.0	485.3
17	4	2	5	3	1	4	61.8	376.5	461.3
18	4	3	1	4	2	5	166.2	376.9	410.2
19	4	4	2	5	3	1	190.2	376.5	383.9
20	4	5	3	1	4	2	101.6	375.8	380.0
21	5	1	5	4	3	2	122.0	377.9	493.0
22	5	2	1	5	4	3	123.8	379.2	406.0
23	5	3	2	1	5	4	117.2	376.4	376.4
24	5	4	3	2	1	5	143.0	378.8	385.7
25	5	5	4	3	2	1	182.4	376.9	386.3
K_{11}	1 890.5	1 885.8	1 888.2	1 882.9	1 889.7	1 886.3	—	—	—
K_{12}	1 884.5	1 886.1	1 884.3	1 889.0	1 884.8	1 887.7	—	—	—
K_{13}	1 884.0	1 889.4	1 886.3	1 885.4	1 886.1	1 888.7	—	—	—
K_{14}	1 882.7	1 886.5	1 886.6	1 886.1	1 887.2	1 884.0	—	—	—
K_{15}	1 889.2	1 883.1	1 885.5	1 887.5	1 883.1	1 884.2	—	—	—
R_1	7.8	6.3	3.9	6.1	6.6	4.7	—	—	—

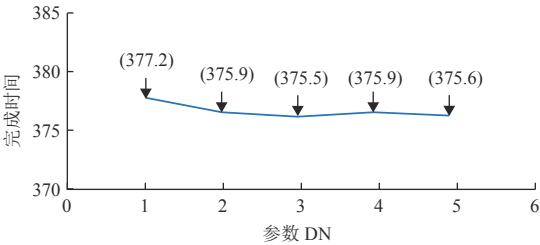


图 8 参数DN在之前所获得的最优的参数的基础上对其进行讨论

Fig.8 Parameters discussion on the basis of the optimal parameters obtained before

的交叉子对遗传算法的收敛性有所改善。其曲线所代表的种群内各个体的 $C_{\max}$ 值(适应度的绝对值)的

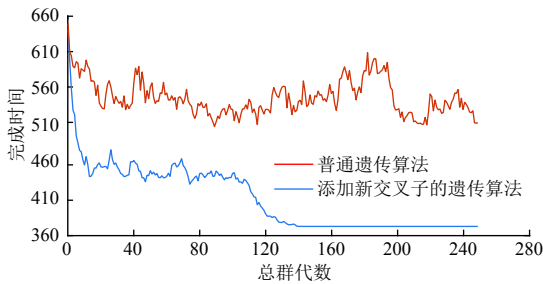


图 9 在250次迭代过程中，种群内个体的均值变化曲线对比

Fig.9 Comparison of mean change curves of individuals in the population during 250 iterations

均值在大约140代收敛于最优值，收敛性明显提高。普通的交叉子由于HFS问题的巨大规模再加上其轮

赌式的交叉规则, 使在个体产生细微变化就会导致适应度大幅变化的环境下, 却实行无方向式的概率交配。通过给改进遗传算法添加改进的交叉算子, 提高了总群后代继承前代优秀基因片段的能力, 让算法达到更好的收敛效果。

图10代表当前迭代次数出现过的最优个体的 $C_{\max}$ 。蓝色曲线代表添加了最优个体邻域搜索, 而且使用了具备模拟退火算法(SAA)机制变异子的改进遗传算法, 而棕色曲线则代表普通遗传算法。从图10可知, 添加了最优个体邻域搜索, 而且使用改进的变异子对遗传算法的搜索能力有所提高。对比普通的遗传算法, 可以看出蓝色曲线的最优 $C_{\max}$ 值为373在大约第77代获得, 而棕色曲线的最优 $C_{\max}$ 值为390在第87代获得, 算法的搜索能力有明显的提高。通过给改进遗传算法添加改进的变异算子, 提高了群内个体探索新基因组合的能力; 通过给改进遗传算法添加邻域搜索, 提高了最优个体主动探索的能力, 让算法达到更好的搜索效果。

5.3 效果对比

表4为相同MPN模型下相同案例中, 采用文献[11]中的蚁群优化算法、标准遗传算法和采用本文的改进遗传算法进行对比所得的结果。采用改进遗传算法所得出的最优解373明显优于文献[11]蚁群算法中所得出的最优解393与标准遗传算法的523, 而且达到最优的次数在100次中出现了83次, 优化结果稳

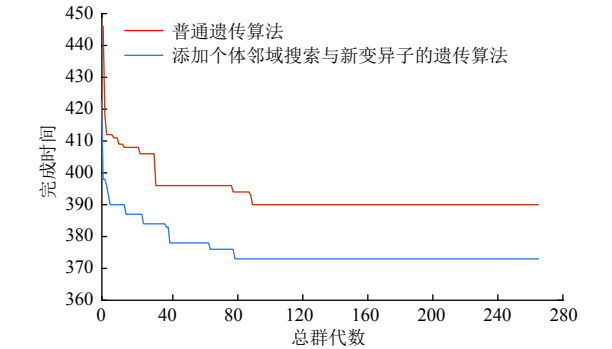


图 10 在 250 次迭代过程中, 最优解的变化过程对比。
Fig.10 Comparison of the change process of the optimal solution during 250 iterations.

定。而文献[11]中的蚁群优化达到最优的次数有 79 次, 虽然较为稳定但最优解却只有 393。标准遗传算法因为无法适应大规模的混流车间问题, 在最优解上只得到了 523, 到达的次数也只有 25 次。从中可以看出改进遗传算法相比于蚁群优化算法与标准遗传算法能够获得相对更优的优化结果且达到最优解更为稳定。如图11为案例最优结果甘特图。

表 4 效果对比
Table 4 Effect comparison

算法	测试次数	最优解	平均值	达到最优的次数
蚁群优化	100	393	394.7	79
遗传算法	100	523	540.4	25
改进遗传算法	100	373	373.8	83

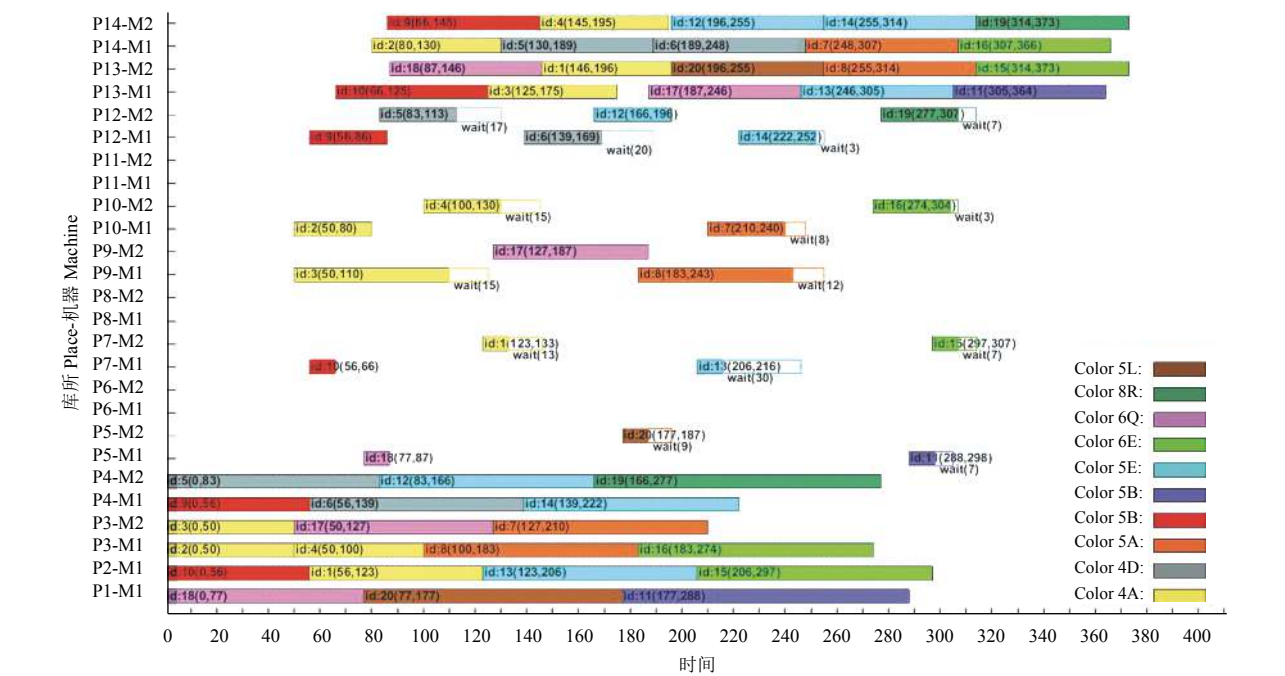


图 11 实验结果甘特图
Fig.11 Gantt chart of experimental results

6 结论

为了解决基于MPN模型下的混流车间问题,本文采用了一种改进的遗传算法对优化的目标进行求解,并通过实验证明其有效性。最后获得的结论如下。

(1) 在每个染色体进行编码时,利用染色体安排段优化算法对染色体进行补全,并采用固定的求解过程让染色体的选择段与适应度进行绑定,可以有效压缩遗传算法的搜索空间,为HFS大规模问题的求解提供一个缩减后的解空间。

(2) 改进式遗传算法通过染色体进行交叉与变异时,分别使用粒子群与模拟退火的机制,可以提高遗传算法的搜索效率,为求解大规模HFS问题提供优质个体。

(3) 改进式遗传算法通邻域搜索让当代代所得的最优染色体(优质染色体)主动对自己进行优化,实现求解大规模HFS问题的精确搜索。

参考文献:

- [1] GUPTA J N D. Two-stage, hybrid flowshop scheduling problem[J]. *Journal of the Operational Research Society*, 1988, 39: 359-364.
- [2] 何军红, 马国伟, 刘赛, 等. MES柔性作业车间调度优化算法的研究[J]. *工业仪表与自动化装置*, 2020(1): 26-32.
HE J H, MA G W, LIU S, *et al.* Research on MES flexible job shop scheduling optimization algorithm [J]. *Industrial Instrumentation & Automation*, 2020(1): 26-32.
- [3] 王秀萍. 一种面向柔性作业车间调度的启发式算法[J]. *电脑知识与技术*, 2017, 13(23): 216-218.
- [4] 何涛. 面向智能制造的柔性车间调度算法研究[J]. *生产力研究*, 2019(8): 132-135, 154.
HE T. Research on flexible shop scheduling algorithm for intelligent manufacturing [J]. *Productivity Research*, 2019(8): 132-135, 154.
- [5] 黄亚平, 王万良, 熊婧. 基于改进蚁群算法的智能调度系统设计与开发[J]. *浙江工业大学学报*, 2010, 38(3): 251-256.
HUANG Y P, WANG W L, XIONG J. Design and implementation of intelligent scheduling system based on improved ant colony algorithm [J]. *Journal of Zhejiang University of Technology*, 2010, 38(3): 251-256.
- [6] 田松龄, 陈东祥, 王太勇, 等. 一种异步蚁群算法求解柔性作业车间调度问题[J]. *天津大学学报*, 2016, 49(9): 920-928.
TIAN S L, CHEN D X, WANG T Y, *et al.* An asynchronous parallel ant colony optimization for flexible job-shop scheduling problem [J]. *Journal of Tianjin University*, 2016, 49(9): 920-928.
- [7] 白俊峰, 贾志浩, 白一辰. 基于改进遗传算法的车间调度问题研究[J]. *现代制造技术与装备*, 2019(12): 198-200.
BAI J F, JIA Z H, BAI Y C. Research on job shop scheduling problem based on improved genetic algorithm [J]. *Modern Manufacturing Technology and Equipment*, 2019(12): 198-200.
- [8] 余璇, 梁工谦, 董仲慧. 基于混合遗传禁忌搜索算法的多目标柔性作业车间调度[J]. *机械制造*, 2016, 54(8): 90-93.
YU X, LIANG G Q, DONG Z H. Multi-objective & flexible job-shop scheduling based on hybrid genetic tabu search algorithm [J]. *Machinery*, 2016, 54(8): 90-93.
- [9] 吴大立, 郑中祥, 尹项根, 等. 基于Petri网和多种群遗传算法的海洋核动力平台电力系统网络重构[J]. *电力自动化设备*, 2020, 40(8): 160-166.
WU D L, ZHENG Z X, YIN X G, *et al.* Network reconstruction of offshore nuclear power platform power system based on Petri net and multi-population genetic algorithm [J]. *Electric Power Automation Equipment*, 2020, 40(8): 160-166.
- [10] 田海霖, 洪良, 王艺翔, 等. 基于量子遗传算法优化粗糙-Petri网的电网故障诊断[J]. *西安工程大学学报*, 2018, 32(6): 678-684.
TIAN H L, HONG L, WANG Y X, *et al.* Optimization of power grid fault diagnosis of rough-Petri network based on quantum genetic algorithm [J]. *Journal of Xi'an Polytechnic University*, 2018, 32(6): 678-684.
- [11] WANG M L, ZHONG R Y, DAI Q Y, *et al.* A MPN-based scheduling model for IoT-enabled hybrid flow shop manufacturing [J]. *Advanced Engineering Informatics*, 2016, 30(4): 728-736.
- [12] GHOLAMI O, SOTSKOV Y N, WERNER F, *et al.* A genetic algorithm for hybrid job-shop scheduling problems with minimizing the makespan or mean flow time [J]. *Journal of Advanced Manufacturing Systems*, 2018, 17(4): 461-486.
- [13] LIN C S, LEE I L, WU M C, *et al.* Merits of using chromosome representations and shadow chromosomes in genetic algorithms for solving scheduling problems [J]. *Robotics and Computer-integrated Manufacturing*, 2019, 58: 196-207.
- [14] 张国辉, 胡一凡, 孙靖贺. 改进遗传算法求解多时间约束的柔性作业车间调度问题[J]. *工业工程*, 2020, 23(2): 19-25.
ZHANG G H, HU Y F, SUN J H. An improved genetic algorithm for flexible job shop scheduling problem with multiple time constraints [J]. *Industrial Engineering Journal*, 2020, 23(2): 19-25.
- [15] 谢志强, 张伟涛, 杨静. 前移存在调整时间综合调度工序的算法[J]. *机械工程学报*, 2012, 48(12): 169-177.
XIE Z Q, ZHANG W T, YANG J. Algorithm of moving integrated scheduling procedures with set-up time forward [J]. *Journal of Mechanical Engineering*, 2012, 48(12): 169-177.
- [16] DAI M, TANG D, GIRET A, *et al.* Multi-objective optimization for energy-efficient flexible job shop scheduling problem with transportation constraints [J]. *Robotics & Computer Integrated Manufacturing*, 2019, 59: 143-157.

(责任编辑: 杨耀辉)