

## 改进NSGA算法求解多目标柔性车间作业调度问题

鞠录岩<sup>1</sup>, 杨建军<sup>2</sup>, 张建兵<sup>1</sup>, 郭龙龙<sup>1</sup>, 李锁斌<sup>1</sup>

1. 西安石油大学 机械工程学院, 西安 710065

2. 青岛理工大学 机械与汽车工程学院, 山东 青岛 266520

**摘要:** 在多目标柔性车间作业调度问题的研究中, 求解算法与多目标处理至关重要。因此, 基于非支配排序遗传算法提出了改进遗传算法求解该问题, 设计了相应的矩阵编码、交叉算子, 改进了非劣前沿分级方法, 并提出了基于Pareto等级的自适应变异算子以及精英保留策略。实例计算表明, 该算法可以利用传统遗传算法全局搜索能力的同时可以防止早熟现象的发生。改进非劣前沿分级方法可以快速得到Pareto最优解集, 进一步减小了计算复杂度, 而且可以根据种群的多样性改变变异概率, 有利于保持种群多样性、发掘潜力个体。

**关键词:** 柔性车间作业调度; 多目标优化; 非劣前沿分级遗传算法

**文献标志码:** A **中图分类号:** TP301 **doi:** 10.3778/j.issn.1002-8331.1809-0246

鞠录岩, 杨建军, 张建兵, 等. 改进NSGA算法求解多目标柔性车间作业调度问题. 计算机工程与应用, 2019, 55(13): 260-265.

JU Luyan, YANG Jianjun, ZHANG Jianbing, et al. Improved NSGA for multi-objective flexible job-shop scheduling problem. Computer Engineering and Applications, 2019, 55(13): 260-265.

### Improved NSGA for Multi-Objective Flexible Job-Shop Scheduling Problem

JU Luyan<sup>1</sup>, YANG Jianjun<sup>2</sup>, ZHANG Jianbing<sup>1</sup>, GUO Longlong<sup>1</sup>, LI Suobin<sup>1</sup>

1. College of Mechanical Engineering, Xi'an Shiyou University, Xi'an 710065, China

2. School of Mechanical and Automotive Engineering, Qingdao University of Technology, Qingdao, Shandong 266520, China

**Abstract:** During the evaluation process of the job-shop scheduling problem, the algorithm and multi-objective optimization are very important. Therefore, an improved genetic algorithm based on NSGA is proposed and the corresponding matrix coding, decoding and crossover operators are designed. To reduce the computational complexity and improve the performance of the algorithm, a novel non-dominated sorting method, adaptive mutation operators and elite retention strategies are introduced. The simulation experiments show that this non-dominated sorting method can get the Pareto optimal solutions quickly and correctly by dividing the whole population into three parts. This algorithm can make full use of the global searching ability of traditional genetic algorithm, prevent the occurrence of precocious phenomenon, and change the mutation probability according to the diversity of the population.

**Key words:** flexible job-shop scheduling problem; multi-objective optimization; non-dominated sorting genetic algorithm

### 1 引言

制造业产值约占世界生产总值的18%, 对世界对经济发展起到至关重要的作用。提高制造效率, 有助于提高企业的利润率、市场占有率、新产品研发周期等, 是提高企业竞争力的必要手段。柔性车间作业调度(Flexible Job-shop Scheduling Problem, FJSP)是制造

业信息化的重要内容, 是指 $N$ 个工件要在 $M$ 台机床上加工, 每个工件包含一道或多道工序, 工序顺序是预先确定的, 每道工序可以在多台不同的机床上加工, 且工序的加工时间随着机床的不同而不同。FJSP问题突破了传统车间作业调度问题(Job-shop Scheduling Problem, JSP)加工机床唯一的限制, 每道工序可以由不同的

**作者简介:** 鞠录岩(1986—), 男, 博士, 讲师, 主要研究领域为生产计划与调度理论、方法及应用的研究, E-mail: jly171207@xsyu.edu.cn; 杨建军(1977—), 男, 博士, 副教授, 主要研究领域为系统工程优化。

**收稿日期:** 2018-09-19 **修回日期:** 2018-11-01 **文章编号:** 1002-8331(2019)13-0260-06

**CNKI网络出版:** 2018-12-29, <http://kns.cnki.net/kcms/detail/11.2127.TP.20181227.1800.052.html>

机床加工,更符合生产实际。由于FJSP问题的复杂性、随机性、多约束性、多目标性,且实际生产环境中存在着多变性和随机事件,是一个典型的NP难题,如何求得满足要求的准最优解是一个急待解决的问题,所以对它的研究具有重要的现实意义<sup>[1-2]</sup>。

近年来,生物学、人工智能、神经网络、计算机技术及仿真技术的发展为FJSP问题的研究开辟了新的思路<sup>[3-5]</sup>。遗传算法(Genetic Algorithm, GA)对求解作业车间调度问题具有较好效果,Amjad<sup>[6]</sup>通过研究发现,近20年关于FJSP问题的研究中有大约26.4%采用了GA方法求解,GA方法被认为是解决FJSP问题最有效的进化算法之一。但是经典GA中交叉概率和变异概率的选择是影响算法行为和性能的关键,直接影响算法收敛性。因此,很多学者对GA做出了改进以提高算法的求解效率。Li<sup>[7]</sup>提出了一种混合遗传算法,充分利用了遗传算法的全局搜索能力和禁忌搜索算法的局部搜索能力,对求解FJSP问题具有有效性和提高搜索效率。Chen<sup>[8]</sup>提出了一种基于GA的启发式算法,将工序编码与机床编码分开进行,并分别进行交叉和变异操作,以加快收敛速度。Cinar<sup>[9]</sup>提出了一种基于优先权的GA,为了获得改进的解决方案,将迭代局部搜索(ILS)应用于每个再现过程结束时的染色体。

为了解决多目标FJSP优化问题,Liu等<sup>[10-12]</sup>提出了任务粒度定量分析方法,可以准确地评价集成制造系统中复杂产品协同工作流的任务粒度,并用该方法指导粗粒度任务的分解和重构,取得了很好的效果。此外,权重和法、效用函数法、妥协法、字典排序法和Pareto方法等也得到较为广泛的关注和研究<sup>[13-16]</sup>。Srinivas和Deb<sup>[17]</sup>提出了基于Pareto最优解的非劣前沿分级遗传算法(NSGA),在多目标优化领域表现出了非常强的优势,但是其仍旧存在计算复杂度高、缺少保优策略等不足。随后,一些学者对NSGA做了进一步研究,Deb等人<sup>[18]</sup>对NSGA做了改进,引入了精英保留策略和拥挤度计算,提出了NSGA-II算法,提高了算法的性能。张超勇<sup>[19]</sup>提出了层次分析法,设计了一种改进的非支配排序遗传算法。

本文在考虑工件运输时间的前提下,以最大完工时间、机床最大负荷、机床总负荷和交货期为目标,对FJSP优化问题进行研究。在NSGA算法的基础上设计相应的矩阵编码、解码、交叉算子,提出了一种全新的非劣前沿分级方法以及基于Pareto等级的自适应变异算子以及精英保留策略。本文算法在保持种群多样性、发掘潜力个体的前提下,进一步减小计算复杂度,提高算法性能。

## 2 问题描述

在现有对于FJSP问题的研究过程中,为了便于分

析往往对车间调度模型进行了许多假设,这些假设对实际生产环境作了很大的简化,通常都忽略了非加工时间,这使得车间调度的研究具有可行性,但是同时也使得车间调度的研究成果具有很强的理论性,使得调度的结果不适应于实际生产活动,或者说只适用于特定的环境条件下,不具有通用性。实际生产中,有些非加工时间要考虑机床、工序之间的先后顺序约束,无法为加工时间的一部分需要单独考虑,比如工件的运输时间。

考虑工件运输时间的FJSP调度问题的数学模型定义如下:有 $M$ 台不同的机床,加工 $N$ 个工件,每个工件有 $n_j$ 道工序,工序的加工顺序已定,且每道工序可以选用不同的机床加工(不同机床对同一工序的加工时间不同);同一工件相邻两道工序在不同机床上加工时,需要有一个运输时间,且不同机床之间的运输时间不同。在满足工艺约束的同时,FJSP问题还假设满足如下约束:

- (1)同一时刻一台机器只能加工一个工件;
- (2)同一时刻一个工件只能在一台机器上加工;
- (3)任何工件没有优先加工的权利;
- (4)工件一旦开始加工就不能被中断;
- (5)加工时间与机床之间的运输时间已知,且固定,

其他非加工时间如工件准备时间、机床调整时间、等待时间等不予考虑。

本文综合考虑最大完工时间、机床最大负荷、机床总负荷和交货期,建立了在 $M$ 台机床上加工 $N$ 个工件的4目标FJSP最小值为最优解的数学模型,如公式(1)所示:

$$\min[f_1(x), f_2(x), f_3(x), f_4(x)] \quad (1)$$

$$f_1 = \max C_i \quad (2)$$

$$f_2 = \left\{ \sum_{i=1}^N \sum_{j=1}^{n_j} s_{ij} x_{ijk} \right\} \quad (3)$$

$$f_3 = \sum_{k=1}^M \sum_{i=1}^N \sum_{j=1}^{n_j} s_{ij} x_{ijk} \quad (4)$$

$$f_4 = \sum_{i=1}^N \alpha_i (C_i - d_i) \quad (5)$$

式中 $f_1$ 表示最大完工时间; $f_2$ 表示机床最大负荷; $f_3$ 表示机床总负荷; $f_4$ 表示工件延期交货的惩罚。 $i=1, 2, \dots, N$ 表示工件; $j=1, 2, \dots, n_j$ 表示工序; $k=1, 2, \dots, M$ 表示机床。 $C_i$ 为工件 $i$ 的完工时间; $s_{ij}$ 为工件 $i$ 第 $j$ 道工序的加工时间; $x_{ijk}$ 为工件 $i$ 的第 $j$ 道工序是否在第 $k$ 台机床上加工,如果是则为1,否则为0; $\alpha_i$ 为工件 $i$ 延误完工的惩罚系数; $d_i$ 为工件 $i$ 的交货期。

## 3 改进Pareto分级遗传算法设计

### 3.1 编码设计

在对FJSP问题进行编码时不但要考虑工序排列,

还要考虑每道工序选择的机床是否为可选机床。由于工件的运输时间没有被忽略,所以染色体中还需要带有工序加工时间及工件运输时间,而且解码算子在计算工序的完工时间时就需要考虑相同工件的相邻两道工序之间是否在同一台机床上加工,如果在同一台机床上加工则不需要搬运,运输时间设为0,如果不在同一台机床上加工就需要计算运输时间,并比较运输完成的时间与加工机床完工时间,取较大的值作为工序的完工时间。为了解决上述问题,本文采用基于工件的矩阵编码方法,采用一个二维矩阵  $S_{[4][n \times n_j]}$  进行编码。第一行为工序编码层,第二行为机床编码层,第三行为加工时间码层,第四行为运输时间码层,如表1所示。其中工序编码与机床编码为显性基因,参与交叉、变异操作,加工时间编码与运输时间编码为隐性基因,不需要参与交叉、变异,但是需要根据交叉变异结果重新编码。

表1 染色体编码表

		基因编码											
$P_1$	4	1	3	4	2	1	2	3	3	1	2	4	
	3	1	2	6	5	2	2	4	5	1	3	1	
	7	2	6	5	2	3	3	3	9	1	4	1	
	0	0	0	4	0	5	4	5	1	5	3	2	
$P_2$	1	3	2	3	4	1	4	2	2	3	4	1	
	1	1	5	5	3	2	6	1	3	5	1	1	
	2	5	2	5	7	3	5	4	4	9	1	1	
	0	0	0	3	0	5	4	3	7	0	2	5	

工序编码层采用基于工件的编码方式,即每一个基因代表一个工件,每个工件出现的先后顺序代表工序,如一个  $N$  个工件每个工件有  $n_j$  道工序的 FJSP 问题,共有  $N \times n_j$  道工序,染色体含有  $N \times n_j$  个基因,每个基因由1到  $N$  的数字构成,每个数字会出现  $n_j$  次,按照出现的先后顺序分别代表工序1到  $n_j$ ,保证了生成染色体的有效性。

机床编码层、加工时间编码层和运输时间编码层分别采用基于整数的编码方式。机床码是由机床编号组成的,根据工序编码层的基因随机选择对应工序的可选加工机床,将机床编码填入机床编码层的对应位置,使得生成的机床编码均为有效编码。加工时间码是根据工序码和机床码计算得到,运输时间码则是由工序码、机床码和加工时间码计算得到。

表1所示的两个染色体,表示4个工件在6台机床加工且每个工件包含3道工序,每道工序最多有6台可选的加工机床的 FJSP 问题的编码。染色体  $P_1$  的第四列基因表示工件4的第2道工序由机床6加工,加工时间为5,第1道工序加工完后需要经过4的运输时间将工件从机床3运输到机床6。这种编码方式可以保证染色体编码带有足够的基因信息,能保证产生解的完备性与合法性,符合编码原则的要求。

### 3.2 多目标处理、选择操作与精英保留策略

为了便于描述以下介绍一下 Pareto 法的基本概念: Pareto 最优解(又称非受支配解),对于多目标最小化问题,若对于  $\forall_i$  都满足  $f_i(x) \geq f_i(y)$ ,且  $\exists j$ ,使得  $f_j(x) > f_j(y)$ ,则称解  $y$  支配解  $x$ ,记作  $y > x$  即  $y$  相对于  $x$  有优势, $y$  为非受支配解, $x$  为受支配的解;反之,记作  $y < x$ ;若在解空间中任意解均不能支配解  $x$ ,则称解  $x$  为 Pareto 最优解或非受支配解。一系列 Pareto 最优解的集合,称为 Pareto 边界。

为了求解 Pareto 最优解集就要对所有的解进行非劣前沿分级,目前应用最广泛的分级方法如下:假设  $P$  为  $n$  个目标的进化群体,初始时令非支配集  $N = \Phi$ ,  $H$  为非支配解集合中解的个数,初始  $H = 0$ 。

(1)从进化群体  $P$  中选一个体  $X$ 。

(2)令  $P = P - X$ ,  $N = N + X$ ,  $H = H + 1$ 。

(3)若  $P$  为空,则执行步骤(6),否则从  $P$  中任取一个体  $Y$ ,令  $P = P - Y$ ,取  $H' = H$ 。

(4)从  $N$  中取任  $Y$  一个体  $Z$ ,令  $H' = H' - 1$ 。

(5)若  $Z > Y$ ,则执行步骤(3);若  $Z < Y$ ,则  $H = H - 1$ ,  $N = N - Z$ ;若  $H' = 0$ ,则  $N + Y$ ,  $H = H + 1$ ,执行步骤(3);否则执行步骤(4)。

(6)输出非支配解集  $N$  中的解。

上述算法中每个个体都与集合  $P$  中的个体比较,尽管集合  $P$  中的随着分级的进行个体数量在减少,但是仍旧计算量非常大。本文设计了一种新的非劣前沿分级方法,下面给出该方法的具体步骤:

(1)设置非劣前沿等级号  $No = 1$ ,将种群存入集合  $S$  中。

(2)在  $S$  中随机选择一个体  $i$ ,分别找出相对于该个体的非受支配解、受支配解和处于同一非劣前沿级的解,分别存放在  $Q_1$ 、 $Q_2$ 、 $Q_3$  三个集合中。

(3)如果  $Q_1$  不为空,则在  $Q_1$  中选一个体  $j$ ,执行第(2)步;否则在  $Q_2$  中依次寻找每一个个体的非受支配解,如果找不到则将该个体的非劣前沿等级号设为  $N$ ,并存放在  $Q_0$  集合中,然后从  $S$  中去除该个体;如果找到则将该个体的非受支配解存放在  $Q_1$  中。

(4)非劣前沿等级号  $No$  加1。

(5)如果  $Q_1$ 、 $Q_2$  为空则执行第(2)步,直至  $Q_3$  为空,停止运算;否则执行第(3)步。

本文提供的非劣前沿分级方法每次比较均在非受支配解中进行,计算量大大减少,当且仅当每次选取的个体全部是最劣解时,其计算量与上述方法相同,所以此方法可以有效地减少非劣前沿分级时的计算量。

非劣前沿分级完毕以后每一解都具有一个非劣前沿等级号,越优的解其非劣前沿的等级号越小,就可以根据非劣前沿等级号的大小进行选择。随机从种群中



选择两个个体,如果它们处于不同的非劣前沿级别,则选择级别号较小的复制到子代;如果它们处于同一级别,则根据小生境技术,选择具有较小拥挤度的个体复制到子代。第1代第 $t$ 个目标函数的最佳小生境尺寸计算方法如下:

$$\sigma_{lt} = \frac{\max f_{lt} - \min f_{lt}}{\sqrt{n \cdot Pop\_size}}, l = 1, 2, \cdots, N \tag{6}$$

式中, $f_{lt}$ 为第 $l$ 代第 $t$ 个目标函数的函数值; $n$ 为目标函数个数; $Pop\_size$ 为群体规模。

为了保证产生的最优解能顺利的遗传到子代中,选择非劣前沿等级为1的个体直接复制到子代中,如果非劣前沿等级为1的个体超过了种群总数的5%,则选择总种群数量5%的不同个体复制到子代,以实现优良基因的保护。

3.3 交叉与变异

由于本文设计了矩阵编码,所以交叉算子需要考虑到工序的变动和机床的变动,针对FJSP优化问题中机床是可选的情况,本文设计的交叉算子分为如下两部分。

(1)工序交叉:工序交叉操作时,采用基于工件顺序的交叉方法,交换两个父代染色体中交叉工件在染色体中位置,各工序的加工机床不做改变。由于交叉操作中工序与机床不变,仅对工序码在染色中的位置做了调整,因此交叉完成后只需要重新计算一下运输时间即可得到有效染色体。如染色体 $P_1$ 和 $P_2$ 进行交叉,随机选取工件1作为交叉工件,交换 $P_1$ 和 $P_2$ 中工件1各工序在染色体中的位置,得到子代 $C_1$ 、 $C_2$ 如表2所示。

		基因编码											
$C_1$	1	4	3	4	2	1	2	3	3	2	4	1	
	1	3	2	6	5	2	2	4	5	3	1	1	
	2	7	6	5	2	3	3	3	9	4	1	1	
	0	0	0	4	0	5	4	5	1	3	2	5	
$C_2$	3	1	2	3	4	1	4	2	2	1	3	4	
	1	1	5	5	3	2	6	1	3	1	5	1	
	5	2	2	5	7	3	5	4	4	1	9	1	
	0	0	0	3	0	5	4	3	7	5	0	2	

(2)机床交叉:机床交叉操作时,保持各工序不变,对机床码进行多点交叉。首先随机产生一个由0、1代码组成的交叉编码串,其长度与染色体相等,然后找到 $P_1$ 染色体中交叉编码串中1对应位置的相应工序,并与 $P_2$ 染色体中对应工序的机床码互换。由于交换的是相同工件的相同工序的加工机床,交换后的机床均为相应工序的可选择机床,所以只需对运输时间码重新编码即可得到有效染色体。如取上述父代 $P_1$ 和 $P_2$ ,随机产生的交叉编码串为(0,0,1,0,0,0,1,0,0,0,0,0),交换工序 $o_{31}$ 和 $o_{22}$ 与的加工机床得到的子代 $C_3$ 和 $C_4$ 如表3所示。

		基因编码											
$C_3$	4	1	3	4	2	1	2	3	3	1	2	4	
	3	1	1	6	5	2	1	4	5	1	3	1	
	7	2	5	5	2	3	4	3	9	1	4	1	
	0	0	0	4	0	5	3	10	1	5	7	2	
$C_4$	1	3	2	3	4	1	4	2	2	3	4	1	
	1	2	5	5	3	2	6	2	3	5	1	1	
	2	6	2	5	7	3	5	3	4	9	1	1	
	0	0	0	4	0	5	4	4	3	0	2	5	

在进化初期需要较小的变异概率以保证优良基因不被破坏,进化后期需要较大的变异概率防止“早熟”,因此,根据Pareto对种群分级的特点,第 $l$ 代的变异概率 $p_m$ 为:

$$p_m = k_1 + k_2 \frac{(p_s - N_l)}{p_s} \tag{7}$$

其中 $k_1$ 、 $k_2$ 为(0~1)调节系数; $P_s$ 为种群大小; $N_l$ 为第 $l$ 代的Pareto总级数。在种群多样性好的时候Pareto总级数会比较大,此时的变异概率比较小,当种群的多样性下降时Pareto总级数也会相应的减小,变异概率会增大以便于产生新的基因增加种群多样性。

由于染色体采用矩阵编码方式,所以变异算子也分为工序变异和机床变异。工序变异时各工序的加工机床不变,随机选择染色体两个不同工件的两个基因交换其在染色体上的位置;机床变异时随机选择一道工序,然后选择一台可选机床替换其原始加工机床,并重新计算运输时间。

4 仿真结果与分析

为了测试本文算法的求解能力,使用与刘晓霞、张超勇相同的4×6算例<sup>[20]</sup>和10×10算例<sup>[21]</sup>数据进行测试。由于他们的数据没有考虑运输时间,所以本文为其添加机床与机床之间的运输时间如表4。将上述改进遗传算法用C++编程实现,采用与参考文献中相同的种群数量,机床交叉概率和工序交叉概率为0.8,机床变异概率和工序变异概率由式(6)得到,变异调节系数 $k_1=0.2$ , $k_2=0.5$ ,其中工序交叉和机床交叉交替进行,工序变异和机床变异同时进行以增大种群的多样性。

		机器之间的运输时间表									
机器		1	2	3	4	5	6	7	8	9	10
1	0	5	7	10	3	2	3	8	5	2	
2	5	0	3	5	4	3	1	5	2	4	
3	7	3	0	8	2	4	4	4	3	1	
4	10	5	8	0	1	3	2	1	2	4	
5	3	4	2	1	0	2	3	7	5	2	
6	2	3	4	3	2	0	4	2	3	1	
7	3	1	4	2	3	4	0	5	7	2	
8	8	5	4	1	7	2	5	0	2	5	
9	5	2	3	2	5	3	7	2	0	6	
10	2	4	1	4	2	1	2	5	6	0	

在不考虑运输时间的前提下采用本文算法得到了上述算例的最优解集,其中表5和表6分别显示了4×6算例和10×10算例 Pareto 最优解集中的一个解,图1和图2分别是各解的甘特图。可以看出在相同的条件下本文得到的4×6算例最优解与文献中处于相同的 Pareto 等级,最大完工时间、机床最大负荷、总负荷、总拖期分别为17、11、44、5。在10×10算例计算中本文同样得到了与参考文献相同的 Pareto 等级的最优解,最大完工时

表5 4×6算例无运输时间最优解

基因编码											
3	2	1	4	3	2	1	4	1	2	3	4
1	5	1	3	4	2	4	4	1	3	5	1
5	2	2	7	3	3	2	4	1	4	9	1
0	0	0	0	0	0	0	0	0	0	0	0

表6 10×10算例无运输时间最优解

基因编码																													
5-1-7-8-3-9-6-2-4-10-10-4-5-9-2-8-1-6-3-7-10-4-5-9-2-8-6-1-3-7																													
9-1-1-5-10-3-6-1-7-3-2-2-9-7-4-10-2-9-8-3-7-4-4-6-3-2-9-4-7-6																													
2-1-1-2-1-1-2-2-1-1-1-2-1-1-1-4-1-2-1-1-2-1-1-1-2-2-1-1-1-1																													
0-0																													

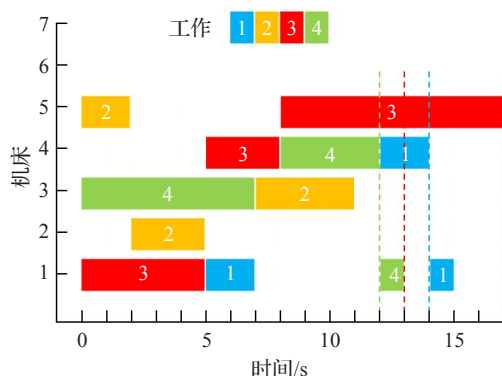


图1 4×6算例无运输时间最优解甘特图

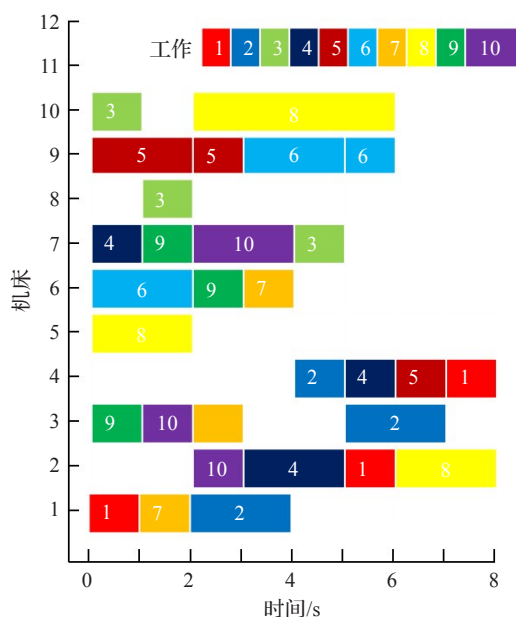


图2 10×10算例无运输时间最优解甘特图

间、机床最大负荷、总负荷、总拖期分别为8、6、42、0。

为了分析种群的进化,分别对两个算例计算200代和600代,在每一代的最优解集中取一个解得到图3和图4。

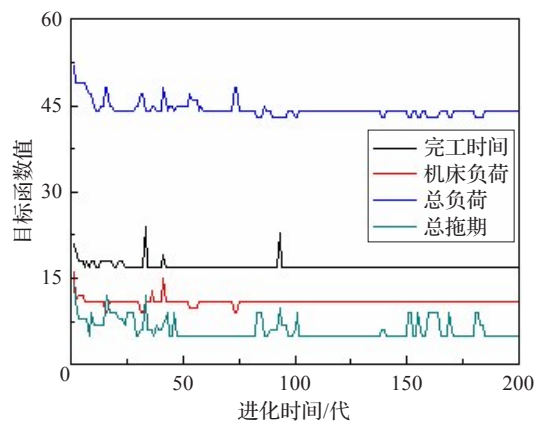


图3 4×6算例种群进化分析图

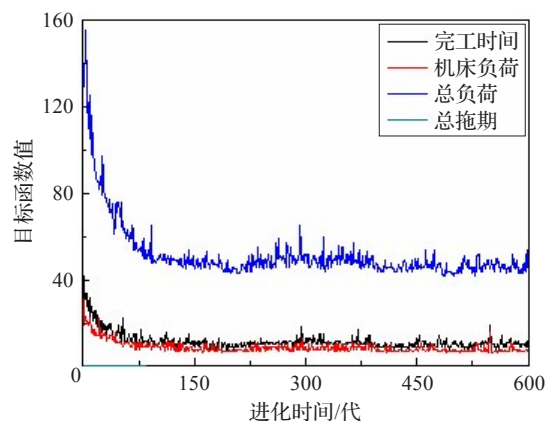


图4 10×10算例种群进化分析图

可以发现,在大约50代的时候本文得到了4×6算例的最优解,与参考文献相同。对于10×10算例大约150代的时候就能搜寻到最优解集,远远小于参考文献中的600代。通过图3和图4可以发现,本文算法在处理多目标FJSP问题时具有较强的搜索能力,可以快速的得到问题的最优解集,并且在得到最优解集之前并没有出现早熟现象,得到最优解后也没有出现种群退化现象。此外,在得到最优解之后,各目标函数的曲线依旧会出现波动,表明在得到最优解集之后本算法仍可以继续寻优,丰富最优解集,体现了较强的搜索能力和计算能力。

通过图1可以看出,在不考虑运输时间的前提下,4×6算例中工件2的第一道工序在机床5上加工,然后第二道工序立即在机床2上加工,中间没有任何工件运输的过程,这违背了实际工况。在考虑运输时间之后得到了4×6算例与10×10算例考虑运输时间的最优 Pareto 解集如表7和表8所示,各目标函数值分别为(22, 16, 49, 13)和(14, 7, 46, 0),图5和图6分别为各解的甘特图。得到的优化结果可以看出,考虑运输时间后,工件

的一道工序加工完成后工件需要运输到位才可以进行下一道工序的加工,造成了最优解的质量有所下降,但是对比甘特图可以看出这样更加贴近生产实际,更具有指导实际生产的作用。

表7 4×6有运输时间最优解

基因编码											
1	4	2	3	1	2	2	1	3	3	4	4
2	6	5	1	2	2	3	1	5	5	6	1
3	8	2	5	3	3	4	1	5	9	5	1
0	0	0	0	0	4	3	5	3	0	0	2

表8 10×10有运输时间最优解

基因编码																			
4	7	3	8	6	1	2	10	5	9	9	5	2	2	8	7	1	3	6	10
4	7	6	4	3	5	9	10	8	1	9	1	10	5	6	1	1	3	9	3
7	9	10	7	3	3	8	9	2	2	6	9	4	7	2	6	8	2	10	
1	1	1	2	2	1	2	1	2	1	1	1	2	4	1	1	1	2	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

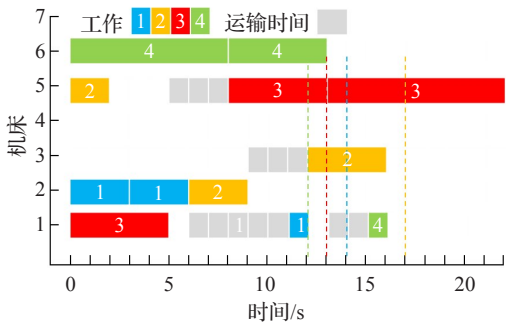


图5 4×6算例有运输时间解甘特图

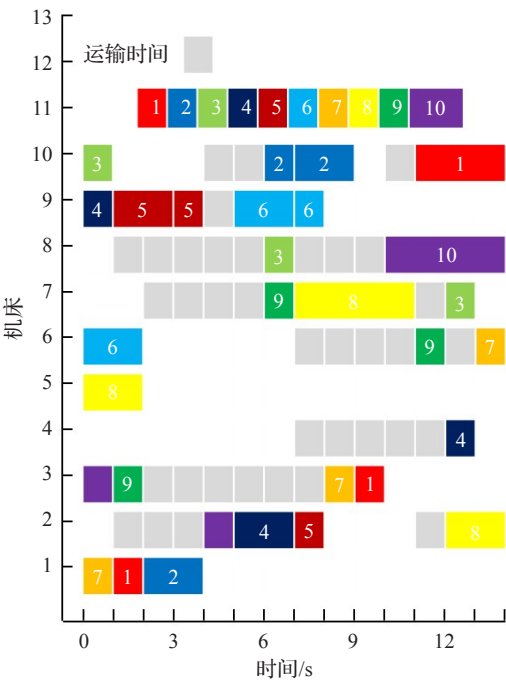


图6 10×10算例有运输时间最优解甘特图

5 结束语

本文在进行多目标FJSP调度优化时考虑了运输时

间对调度结果的影响,比传统JSP调度模型更符合车间调度现状,同时提出了基于NSGA的改进遗传算法求解本问题,得到的主要结论如下:

(1)提出了基于NSGA算法的改进遗传算法,可以有效地求解考虑运输时间的多目标柔性车间作业调度问题。

(2)针对考虑运输时间的多目标柔性车间作业调度的特点,设计的矩阵编码、交叉算子,并改进了非劣前沿分级方法,可以快速得到 Pareto 最优解集,进一步减小了计算复杂度。

(3)提出了基于 Pareto 等级的自适应变异算子以及精英保留策略。可以根据种群的多样性改变变异概率,有利于保持种群多样性,发掘潜力个体。

参考文献:

[1] 薛宏全,魏生民,张鹏,等.基于多种群蚁群算法的柔性作业车间调度研究[J].计算机工程与应用,2013,49(24):243-248.

[2] 王凌,邓瑾,王圣尧.分布式车间调度优化算法研究综述[J].控制与决策,2016,31(1):1-11.

[3] 施文章,韩伟,戴睿闻.模拟退火下布谷鸟算法求解车间作业调度问题[J].计算机工程与应用,2017,53(17):249-253.

[4] Mokhtari H,Hasani A.An energy efficient multi-objective optimization for flexible job shop scheduling problem[J].Comput Chem Eng,2017,104(2):339-352.

[5] 屈迟文,傅彦铭,罗明山,等.求解柔性作业车间调度问题的鸟群算法[J].计算机工程与应用,2018,54(17):249-257.

[6] Amjad M K,Butt S I,Kousar R,et al.Recent research trends in genetic algorithm based flexible job shop scheduling problems[J].Mathematical Problems in Engineering,2018(8):1-32.

[7] Li X,Gao L.An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem[J].Int J Prod Econ,2016,174:93-110.

[8] Chen M,Li J.Genetic algorithm combined with gradient information for flexible job-shop scheduling problem with different varieties and small batches[C]//Proceedings of 2016 the 3rd International Conference on Mechatronics and Mechanical Engineering,2017.

[9] Cinar D,Oliveira J A,Topcu Y I,et al.A priority-based genetic algorithm for a flexible job shop scheduling problem[J].Journal of Industrial & Management Optimization,2017,12(4):1391-1415.

[10] Liu A,Fowler J,Pfund M.Dynamic co-ordinated scheduling in the supply chain considering flexible routes[J].International Journal of Production Research,2016,54(1):322-335.

[11] Liu A,Pfund M,Fowler J.Scheduling optimization of task allocation in integrated manufacturing system based on task decomposition[J].Journal of Systems Engineering and Electronics,2016,27(2):422-433.

(下转第270页)



表3 各条件下计算每条线平均耗时<sub>min</sub>

数据	SAS 10K	SAS 15K	SSD
数据1	20.00	11.00	10.00
数据2	19.80	10.20	9.60
数据3	21.20	11.30	10.30
数据4	22.10	10.50	9.20
数据5	19.60	9.60	8.60
均值	20.54	10.52	9.54

原因就是CPU与内存配置较第二种服务器差所致,这也说明硬盘配置对该类应用影响不大。

综上,为实现该类应用的高效率运行与良好的性价比,可配置较好的CPU与内存,外加SAS 15K存储,也再次证明本文设计的方案具有一定的可行性。

#### 4 结束语

针对石油勘探地震处理数据量增加导致部分应用中间数据对存储的读写需求提升,设计基于SSD、集群NAS多级分层存储的地震资料并行处理方法,并针对高IOPS应用、大数据块并行应用进行算法流程级设计。通过DD存储性能基准测试、IOZONE存储性能基准测试,对SSD、SAS 15K、SAS 10K三类硬盘的读写性能有了清晰的认知;通过实际运行分选作业与逆时偏作业,证明文中设计的相关应用流程具有的一定的实用性。无论三类硬盘基准测试结果,还是具体应用设计方案,都具有一定的指导意义,可以有效地应用于石油勘探领域。

#### 参考文献:

- [1] Armbrust, M, Fox A, Griffith R, et al. Above the clouds: A berkeley view of cloud computing: UCB/EECS-2009-28[R]. University of California Berkeley EECS Department, 2009.
- [2] 李圣强, 李闽峰, 刘桂平, 等. 高性能集群计算系统的构

建[J]. 地震, 2012, 32(1): 144-149.

- [3] 杜金虎, 杨涛, 李欣. 中国石油天然气股份有限公司“十二五”油气勘探发现与“十三五”展望[J]. 中国石油勘探, 2016, 21(2): 1-15.
- [4] 袁刚, 王西文, 雍运动, 等. 宽方位数据的炮检距向量片域处理及偏移道集校正方法[J]. 石油物探, 2016, 55(1): 84-90.
- [5] 娄兵, 姚茂敏, 罗勇, 等. 高密度宽方位地震数据处理技术在玛湖凹陷的应用[J]. 新疆石油地质, 2015, 36(2): 208-213.
- [6] 杨濮源, 金培权, 岳丽华. 一种时间敏感的SSD和HDD高效混合存储模型[J]. 计算机学报, 2012, 35(11): 2294-2305.
- [7] 蔡涛, 鞠时光, 牛德姣. 基于免疫安全存储设备IBSSD的研究与实现[J]. 计算机科学, 2009, 36(1): 101-104.
- [8] 张强. 基于模块化的集群NAS存储系统的研究与实现[D]. 济南: 山东大学, 2013.
- [9] 刘绍刚, 刘明安. 基于NAS的私有存储网络节点的嵌入式设计与实现[J]. 现代电子技术, 2017, 40(10): 79-81.
- [10] 程付超, 苗放, 陈昱. 自适应的分布式文件系统元数据管理模型[J]. 计算机工程与设计, 2014, 35(3): 867-874.
- [11] Latesh Kumar K J. Implementing network file system protocol for highly available clustered applications on network attached storage[C]//Proceedings of the 5th International Conference on Computational Intelligence and Communication Networks, 2013: 496-499.
- [12] 刘红伟, 刘洪, 邹振, 等. 地震叠前逆时偏移中的去噪与存储[J]. 地球物理学报, 2010, 53(9): 2171-2180.
- [13] 陈可洋. 基于高阶有限差分的波动方程叠前逆时偏移方法[J]. 石油物探, 2009, 48(5): 475-478.
- [14] Biardzki C. D. Metabench—A metadata benchmark for distributed file systems[J]. Journal of Supercomputing, 2011, 57(2): 179-188.
- [15] Vasudevan V, Phanishayee A. FAWN: A fast array of wimpy nodes[J]. Communications of the ACM, 2009, 54(7): 101-109.

(上接第265页)

- [12] Liu A, Liu H, Tsai S B, et al. Using a hybrid model on joint scheduling of berths and quay cranes—from a sustainable perspective[J]. Sustainability, 2018, 10(6): 1-15.
- [13] 李峥峰. 多时间因素作业车间调度问题的研究与工程应用[D]. 武汉: 华中科技大学, 2010.
- [14] 刘林, 郑江. 改进生物地理学算法求解柔性作业调度问题[J]. 计算机工程与应用, 2016, 52(18): 228-234.
- [15] 马邦雄, 叶春明. 基于改进量子猫群算法的流水车间调度研究[J]. 计算机工程与应用, 2015, 51(17): 34-37.
- [16] 张华军, 赵金, 罗慧, 等. 基于个人偏好的多目标优化问题目标权重计算方法[J]. 控制与决策, 2014, 29(8): 1471-1476.
- [17] Srinivas N, Deb K. Multi-objective optimization using

nondominated sorting in genetic algorithms[J]. Evolutionary Computation, 1994, 2(3): 221-248.

- [18] Deb K, Agrawal S, Pratap A, et al. A fast elitist nondominated sorting genetic algorithm for multi-objective optimization: NSGA-II[M]. Berlin Heidelberg: Springer, 2000: 849-858.
- [19] 张超勇, 董星, 王晓娟, 等. 基于改进非支配排序遗传算法的多目标柔性作业车间调度[J]. 机械工程学报, 2010, 46(11): 156-164.
- [20] 刘晓霞, 谢里阳, 陶泽, 等. 柔性作业车间多目标调度优化研究[J]. 东北大学学报(自然科学版), 2008, 29(3): 362-365.
- [21] 张超勇, 饶运清, 李培根, 等. 柔性作业车间调度问题的两级遗传算法[J]. 机械工程学报, 2007, 43(4): 119-124.