

直觉模糊集相似度遗传算法求解多目标车间调度问题

徐文婕, 朱光宇[†]

(福州大学 机械工程及自动化学院, 福建 福州 350116)

摘要: 为了提高高维多目标置换流水车间调度问题的求解质量, 提出基于直觉模糊集相似度的遗传算法(similarity of intuitionistic fuzzy sets GA, SIFS_GA). 算法中分别将参考解和Pareto解映射为参考解直觉模糊集和Pareto解直觉模糊集. 计算两个集合之间的直觉模糊相似度, 用以判断Pareto解的优劣. 以直觉模糊集相似度值引导多目标遗传算法进化. 对6个CEC标准测试集与10个流水车间调度测试实例进行仿真实验, 结果表明SIFS_GA算法性能优于常用的多目标优化算法, 且可以有效解决多目标置换流水车间调度问题, 尤其在解决规模较大的问题上是一种有效方法.

关键词: 多目标优化; 置换流水车间调度; 直觉模糊集相似度; 遗传算法

引用格式: 徐文婕, 朱光宇. 直觉模糊集相似度遗传算法求解多目标车间调度问题. 控制理论与应用, 2019, 36(7): 1057 – 1066

DOI: 10.7641/CTA.2018.80232

Genetic algorithm based on similarity of intuitionistic fuzzy sets for many-objective flow shop scheduling problems

XU Wen-jie, ZHU Guang-yu[†]

(School of Mechanical Engineering & Automation, Fuzhou University, Fuzhou Fujian 350116, China)

Abstract: To obtain better solution of many-objective permutation flow-shop scheduling problems (PFSP), a genetic algorithm based on similarity of intuitionistic fuzzy sets (SIFS_GA) is proposed. In this algorithm, reference solution and Pareto solution are mapped into reference solution intuitionistic fuzzy sets and Pareto solution intuitionistic fuzzy sets respectively. The similarity of intuitionistic fuzzy sets between two sets is calculated and adopted to determine the quality of the Pareto solution. The similarity value of intuitionistic fuzzy sets is used as the fitness value of GA to guide the algorithm evolution. Finally, simulation experiments are carried out with 6 CEC benchmark examples and 10 flow shop scheduling test examples to analyze the proposed algorithm. Experimental results show that SIFS_GA can obtain better results than other commonly used many-objective optimization algorithms, and can effectively solve many-objective flow shop scheduling problems, especially in solving the problem of large scale.

Key words: many-objective optimization; permutation flow-shop scheduling; similarity of intuitionistic fuzzy set; genetic algorithm

Citation: XU Wenjie, ZHU Guangyu. Genetic algorithm based on similarity of intuitionistic fuzzy sets for many-objective flow shop scheduling problems. *Control Theory & Applications*, 2019, 36(7): 1057 – 1066

1 引言

车间调度问题涉及范围较为广泛, 包括单机调度问题、流水车间调度问题、作业车间调度问题和开放车间调度问题等. 其中, 约有25%的生产制造系统可简化为流水车间调度问题, 因此流水车间调度问题

(flow-shop scheduling problem, FSP)是研究最广泛的生产调度问题之一^[1]. 针对单目标流水车间调度问题, 王大志等^[2]提出了一种基于多智能体的进化搜索算法; 张先超等^[3]采用了改进的基本量子进化算法求解; Lin等^[4]提出了一种混合离散生物地理学优化(hybrid

收稿日期: 2018-04-03; 录用日期: 2018-09-11.

[†]通信作者. E-mail: zhugy@fzu.edu.cn; Tel.: +86 13805017844.

本文责任编辑: 薛安克.

工信部2016智能制造综合标准化与新模式应用项目(工信部联装(2016)213号), 福建省科技厅科技计划重点项目(2016H0015), 福建省高端装备制造协同创新中心项目(2015A003), CAD/CAM福建省高校工程研究中心开放基金项目(K201704)资助.

Supported by the Intelligent Manufacturing Integrated Standardization and New Model Application Project in 2016 of MIIT ((2016)213), the Key Project of Science and Technology Plan of Fujian Science and Technology Department (2016H0015), the High-end Equipment Manufacturing Collaborative Innovation Center Project of Fujian (2015A003) and the CAD/CAM Engineering Research Center Open Fund of Fujian Provincial University (K201704).

discrete biogeography-based optimization, HDBBO)算法求解;刘刚等^[5]提出了一种基于混合离散人工蜂群算法.针对多个目标的流水车间调度问题,Lu等^[6]提出了一种混合多目标回溯搜索算法求解;王洪峰等^[7]提出了一种基于分解方法的多种群多目标遗传算法,用以求解.综上可知目前对于流水车间调度问题,考虑一个或两个优化目标居多,考虑3个以上的多目标流水车间调度问题较少.一方面由于流水车间调度问题自身的复杂性,另一方面随着目标维数的增加,采用常用的多目标车间调度算法时,非支配解的比例迅速提升,以支配关系作为适应度值难以区分较优的解,导致高维多目标(目标数量 ≥ 3)流水车间调度极具挑战性.但随着企业竞争越来越激烈,决策者在选取调度方案时,需考虑多种因素,如完工时间、拖期成本、空闲时间、成本、库存等,因此对于高维多目标调度问题的研究显得尤为重要.

遗传算法(genetic algorithm, GA)起源于对生物系统所进行的计算机模拟研究.与其他算法相比,遗传算法具有通用性、适应性、扩展性和隐并行性这些独特的特点,在求解复杂的多目标流水车间调度问题的近似最优解方面具有极大的优势,经过多年的研究和实验,证明了遗传算法在求解多目标流水车间调度问题时,具有很好的可行性和有效性.

直觉模糊集是模糊集理论的一种推广,它同时考虑了隶属度、非隶属度和犹豫度,有众多学者对模糊集测量分析展开了研究,将其应用到各个领域之中.Nguyen等^[8]提出了一种直觉模糊集相似/不相似的度量方法,并将其应用在模式识别中.Ananthi等^[9]提出了一种基于区间直觉模糊集的阈值化方法,将其应用在图像分割中.杨进帅等^[10]提出一种直觉模糊遗传算法,求解武器-目标分配问题.王开军等^[11]将模糊数学中的直觉模糊集和灰色模型相结合,设计故障预测的方法.目前,将直觉模糊集相似度的方法应用到车间调度中的国内外研究的前沿文章不多.

置换流水车间调度问题(permutation flow-shop scheduling problem, PFSP)是许多实际生产调度问题的简化模型.因此本文将直觉模糊集相似度的方法与遗传算法相结合,提出基于直觉模糊集相似度的遗传算法(similarity of intuitionistic fuzzy sets GA, SIFS-GA)应用于多目标置换流水车间调度问题中.通过直觉模糊集相似度来探寻Pareto解及各个目标之间的有效信息,有效消除目标数量级和量纲影响并且解决信息丢失的问题.采用直觉模糊集的相似度作为进化算法的适应度来引导算法进化.通过隶属度函数、非隶属度函数以及直觉指数,将Pareto解映射到直觉模糊集中.另构造参考解直觉模糊集.计算Pareto解直觉模糊集和参考解直觉模糊集的相似度,以相似度大小来判断Pareto解的优劣.分别对6个CEC标准测试集与

10个流水车间调度测试实例进行仿真实验,验证了SIFS-GA算法优越性,并且可以有效解决多目标置换流水车间调度问题,尤其在解决规模较大的问题上是一种有效方法.

2 数学模型

置换流水车间调度问题的描述如下: n 个工件在 m 台机器上加工.每个工件的加工路径相同,所有机器上工件的加工次序均相同,一台机器上一个工件的加工时间固定^[12].在不同的机器上不能同时加工同一个工件,多个工件不能同时在一台机器上加工.目标是研究确定所有工件的最优加工序列从而使得设定的性能指标达到最优.其中, n 个工件的集合是 $J = \{1, 2, \dots, n\}$, m 台机器的集合是 $M = \{1, 2, \dots, m\}$.

在收到零售商的订单后,制造商需要加工 n 个工件,加工完成后运送给零售商.受零售商的交付日期和本身库存能力的约束,制造商必须在合理的时间内完成所有的工作.工件提前完成或者拖期完成都会对企业的经济效益造成负面影响.在工件运输之前需要将其存库,如若工件提前完成,将会产生库存成本;如若工件在交货日期之后完成,将会产生拖期成本.库存成本、拖期成本与时间成正比关系.工件加工完成后,将分批运往零售商,出货量的最大运输量是确定并且相同的.当完成的工件的数量达到最大的运输量时,这些工件将被运送,并且每个批次的装运时间是最后一个工件的完工时间.如果最后一批工件的数量少于最大的运输量,最后一批工件仍将作为一批运送.

该问题可通过符号来描述. $x = \{x_1 x_2 \dots x_n\}^T$ 是 n 个决策变量的向量,代表可行的工件加工序列. $f_k(x)$ 是第 k 个目标函数的函数值. $F(x) = (f_1(x) f_2(x) \dots f_k(x) \dots f_M(x))^T$ 称为目标向量或Pareto前沿. $Z^* = (F^*) = (f_1^* f_2^* \dots, f_k^* \dots f_M^*)^T$ 称为理想目标向量,理想目标向量的第 k 个分量是以下问题的最小值: $f_k^* = \min f_k(x)$.其他参数如下:

n : 总的工件数.

m : 总的机器数量.

T_{ik} : 工件 i 在机器 k 上的加工时间($i = 2, \dots, n$; $k = 2, \dots, m$).

$C_{im}(x)$: 在一个工件序列中,工件 i 在机器 m 上的完工时间.

D_i : 第 i 个工件的交货时间.

L_i : 第 i 个工件的出货时间.

β_i : 第 i 个工件单位时间的库存成本.

σ_i : 第 i 个工件单位时间的拖期成本.

$S(j)$: 第 j 批次的实际运输量.

h : 单批次的最大运输量.

b : 运输次数.

在企业中,不同的部门从自身利益出发,会对车间

调度决策给予不同的期望,如生产车间希望提高生产效率,销售部门希望能够准时交货,制造部门希望能尽量降低成本,因此本文设定以下有意义且相互冲突的优化目标:缩短最大完工时间是为了提高企业的生产效率;减短最大延迟时间是为了提高顾客的满意度;降低拖期成本和库存成本是为了提高企业的经济效益.优化的目标函数如下:

最后一个机器上的最后一个工件的完工时间(即最大完工时间)为

$$f_1(x) = \max\{C_{im}(x) | i \in 1, 2, \dots, n\}. \quad (1)$$

由于每个工件的完成延迟,制造商将被收取费用(即最大延迟时间)为

$$f_2(x) = \max\{0, (C_{im}(x) - D_i) | i \in 1, 2, \dots, n\}. \quad (2)$$

运输前每个工件都需要产生库存成本(即总库存成本)为

$$f_3(x) = \sum_{i=1}^n (\beta_i \times \max\{0, (L_i - C_{im}(x))\}). \quad (3)$$

对所有拖期的工件可能会产生一定的拖期成本(即总拖期成本)为

$$f_4(x) = \sum_{i=1}^n (\sigma_i \times \max\{0, (C_{im}(x) - D_i)\}). \quad (4)$$

完工时间 $C_{ik}(x)$ 和目标函数受制于以下约束:

$$C_{11}(x) \geq T_{11}, \quad (5)$$

$$C_{1k}(x) \geq C_{1(k-1)}(x) + T_{11}, \quad k = 2, \dots, m, \quad (6)$$

$$C_{i1}(x) \geq C_{(i-1)1}(x) + T_{i1}, \quad i = 2, \dots, n, \quad (7)$$

$$C_{ik}(x) \geq \max\{C_{i(k-1)}(x), C_{(i-1)k}(x)\} + T_{ik}, \quad i = 2, \dots, n; \quad k = 2, \dots, m, \quad (8)$$

$$S(j) \leq h, \quad j = 1, 2, \dots, b, \quad (9)$$

公式(9)描述了所有运输的运输能力.多目标置换流水车间调度问题的数学模型如下:

$$F(x) = \min\{f_1(x), f_2(x), f_3(x), f_4(x)\}. \quad (10)$$

3 基于直觉模糊集相似度的遗传算法

本文采用直觉模糊集的相似度作为进化算法的适应度来引导算法进化.通过隶属度函数、非隶属度函数以及直觉指数,分别将Pareto解和参考解映射到直觉模糊集中,形成Pareto解直觉模糊集和参考解直觉模糊集.计算两个直觉模糊集之间的相似度,以相似度大小来判断Pareto解的优劣.

3.1 直觉模糊集定义

直觉模糊集(intuitionistic fuzzy set, IFS)^[13]是对模糊集理论的扩充,由Astanassov首次提出,用于处理具有不确定性的不完整信息,它用隶属度、非隶属度及

犹豫度(直觉指数)描述信息来评估比较模糊集间的关系.Astanassov给出的直觉模糊集定义^[13]为

设 X 是元素 x 的一给定有限论域,则 X 上的一个直觉模糊集 A 为

$$A = \{x, \mu_A(x), \gamma_A(x) | x \in X\}, \quad (11)$$

其中:直觉模糊集 A 的隶属度函数为 $\mu_A(x)$,且 $\mu_A(x) : X \rightarrow [0, 1]$; A 的非隶属度函数为 $\gamma_A(x)$,且 $\gamma_A(x) : X \rightarrow [0, 1]$;对于 A 上的所有 $x \in X$,都有 $0 \leq \mu_A(x) + \gamma_A(x) \leq 1$.

为了衡量 x 对 A 的犹豫程度(hesitancy degree),对于 x 中的每个直觉模糊子集,提出 A 中 x 的直觉指数(intuitionistic index)为

$$\pi_A(x) = 1 - \mu_A(x) - \gamma_A(x). \quad (12)$$

若 $\pi_A(x) = 0, \forall x \in X$,则 $\mu_A(x) + \gamma_A(x) = 1$,直觉模糊集合 A 即成为普通模糊集合.

若定义在 X 上的直觉模糊集的全体用 $\text{IFS}(X)$ 表示,那么一个直觉模糊集 $A \in \text{IFS}(X)$,其隶属度 $\mu_A(x)$ 、非隶属度 $\gamma_A(x)$ 以及直觉指数 $\pi_A(x)$ 分别表示对象 x 属于直觉模糊集 A 的支持、反对、中立这三种证据的程度.

3.2 建立直觉模糊集

3.2.1 多目标解直觉模糊集映射

将直觉模糊集相似度应用于多目标优化时,需要将多目标解与直觉模糊集建立联系.为此,本文提出了多目标解直觉模糊集的两步映射,将每个子目标进行 q 次单目标优化,选取 q 次单目标优化时的最大值作为该子目标论域的上界, q 次单目标优化最优值平均值的 λ 倍作为该子目标论域的下界.

第1步 将每一个子目标解映射为直觉模糊子集.

定义映射 H_1 :

$$f_i^j(x) \xrightarrow{H_1} \{x, \mu_{f_i}(j), \gamma_{f_i}(j)\}, \quad (13)$$

式中: $x = [x_1, x_2, \dots, x_n]$ 表示可行的工件加工序列; $i = 1, 2, \dots, M$, M 为目标函数的个数; $j = 1, 2, \dots, N$, N 为解的个数; $f_i^j(x)$ 为第 j 个解的第 i 个子目标函数值; $\mu_{f_i}(j)$ 为第 j 个解的第 i 个子目标函数的隶属度值,表示相似程度; $\gamma_{f_i}(j)$ 为第 j 个解的第 i 个子目标函数的非隶属度值,表示不相似的程度.

其中,定义第 j 个解的第 i 个子目标函数 $f_i^j(x)$ 的隶属度函数如下:

$$\mu_{f_i}(j) = \begin{cases} 1, & f_i^j(x) \leq f_i^{\min}, \\ \frac{f_i^{\max} - f_i^j(x)}{f_i^{\max} - f_i^{\min}}, & f_i^{\min} < f_i^j(x) < f_i^{\max}, \\ 0, & f_i^j(x) > f_i^{\max}, \end{cases} \quad (14)$$

式中: f_i^{\min} 和 f_i^{\max} 为子目标 i 在论域 U 上的上下界,即

各个子目标值的范围. 采用遗传算法对子目标*i*进行单目标优化, 得到*q*次优化后的目标函数平均值 $f_i^0(x)$. 取下界 $f_i^{\min} = \lambda f_i^0(x)$, $\lambda \in (0, 1)$. 取上界 f_i^{\max} 为算法在所有*q*次优化中第1代内该子目标的最大值.

定义第*j*个解的第*i*个子目标函数 $f_i(x)$ 的非隶属度函数如下:

$$\gamma_{f_i}(j) = \begin{cases} 0, & f_i^j(x) \leq f_i^{\min}, \\ \frac{f_i^j(x) - f_i^{\min}}{\alpha_i f_i^{\max} - f_i^{\min}}, & f_i^{\min} < f_i^j(x) < f_i^{\max}, \\ 1, & f_i^j(x) > f_i^{\max}, \end{cases} \quad (15)$$

式中 α_i 为一常数, α_i 一般取 $1 \sim 1.3$ ^[10].

第2步 将每一个子目标的直觉模糊子集的集合映射为多目标直觉模糊集. 根据式(14)和式(15)计算出第*j*个解的第*i*个子目标的直觉模糊子集. 定义映射 H_2 :

$$Y(f_1^j(x), \dots, f_i^j(x), \dots, f_M^j(x)) \xrightarrow{H_2} \text{IFS}(A_{f_1}^j, \dots, A_{f_i}^j, \dots, A_{f_M}^j), \quad (16)$$

式中 $A_{f_i}^j$ 为第*j*个多目标解的第*i*个目标的直觉模糊子集.

综上所述, 为了建立多目标解与直觉模糊集之间的联系, 需要进行 H_1 与 H_2 两步映射. 通过映射 H_1 , 计算出第*j*个解每个子目标的直觉模糊子集. 通过映射 H_2 , 将映射 H_1 得到的*M*个直觉模糊子集映射为直觉模糊集IFS. 映射过程如下:

$$Y(f_1^j(x), \dots, f_i^j(x), \dots, f_M^j(x)) \rightarrow \begin{cases} f_1^j(x) \xrightarrow{H_1} \{x, \mu_{f_1}(j), \gamma_{f_1}(j)\}, \\ \vdots \\ f_i^j(x) \xrightarrow{H_1} \{x, \mu_{f_i}(j), \gamma_{f_i}(j)\}, \\ \vdots \\ f_M^j(x) \xrightarrow{H_1} \{x, \mu_{f_M}(j), \gamma_{f_M}(j)\}, \end{cases} \xrightarrow{H_2}$$

$$q(\text{IFS}(j), \text{IFS}(0)) = \sqrt{\frac{1}{2M} \sum_{i=1}^M [(\mu_{f_i}(j) - \mu_{f_i}(0))^2 + (\gamma_{f_i}(j) - \gamma_{f_i}(0))^2 + \rho \times (\pi_{f_i}(j) - \pi_{f_i}(0))^2]}, \quad (18)$$

式中: $j = 1, 2, \dots, N$, N 为解的个数; ρ 值考虑犹豫状态, 既包含相似, 又包含不相似, 在没有其他先验信息的情况下, 认为相似与不相似各占一半较为合理, 所以取 $\rho = 1/2$.

根据距离测度计算出多目标直觉模糊集之间的相似度:

$$S(\text{IFS}(j), \text{IFS}(0)) = 1 - q(\text{IFS}(j), \text{IFS}(0)). \quad (19)$$

从式(18)和式(19)可以看出, $S \in (0, 1)$. 两个直觉模糊集相似度 $S(\text{IFS}(j), \text{IFS}(0))$ 的值越大, 表明

$$\text{IFS}(x, \mu_{f_1}(j), \gamma_{f_1}(j), \dots, x, \mu_{f_i}(j), \gamma_{f_i}(j), \dots, x, \mu_{f_M}(j), \gamma_{f_M}(j)) = \text{IFS}(A_{f_1}^j, \dots, A_{f_i}^j, \dots, A_{f_M}^j). \quad (17)$$

3.2.2 多目标解直觉模糊集的建立

在多目标优化问题中, 各个目标之间往往是相互矛盾的, 一般不存在使各目标皆达到各自最优值的理想解. 因此在目标空间内, 本文针对各个目标进行*q*次单目标优化, 以*q*次单目标最优值的平均值为分量形成的点, 选作多目标问题的参考解. 为了求解直觉模糊集之间的相似度, 首先需要将参考解和Pareto解都映射为直觉模糊集.

1) 参考解的映射.

采用遗传算法对子目标*i*进行单目标优化, 取*q*次优化后*q*个目标值的平均值组成多目标参考解函数值 $Y_0 = (f_1^0(x), f_2^0(x), \dots, f_M^0(x))$. 通过第3.2.1节的两步映射, 将 Y_0 映射为参考解直觉模糊集 $\text{IFS}(0) = \{A_{f_1}^0, A_{f_2}^0, \dots, A_{f_M}^0\}$. 其中: $f_i^0(x)$ 为第*i*个子目标单目标优化后的平均函数值; $A_{f_i}^0$ 为第*i*个子目标映射后的参考解直觉模糊集, $A_{f_i}^0 = \{ \langle x, \mu_{f_i}(0), \gamma_{f_i}(0) \rangle \}$.

2) Pareto解的映射.

多目标遗传算法每次迭代后, 选出当前的Pareto最优前沿 $Y_j = (f_1^j(x), f_2^j(x), \dots, f_M^j(x))$. 通过两步映射, 将 Y_j 映射为Pareto解直觉模糊集 $\text{IFS}(j) = \{A_{f_1}^j, A_{f_2}^j, \dots, A_{f_M}^j\}$. 其中: $f_i^j(x)$ 为第*j*个Pareto解的第*i*个子目标函数值; $A_{f_i}^j$ 为第*j*个Pareto解的第*i*个子目标映射后的Pareto直觉模糊集, $A_{f_i}^j = \{ \langle x, \mu_{f_i}(j), \gamma_{f_i}(j) \rangle \}$.

3.3 直觉模糊集间的相似性度量

为了计算Pareto解和参考解的相似程度, 需对映射后的直觉模糊集进行相似度测量. 文献[13]根据IFS的几何解释, 提出一种直觉模糊集度量方法. 借鉴文献[14]的研究成果, 本文提出多目标直觉模糊集之间的距离测度公式, 即式(18):

Pareto解直觉模糊集与参考解直觉模糊集越相似, 该Pareto解与参考解越接近, 即Pareto解的质量越好.

3.4 算法流程

为解决置换流水车间调度高维多目标优化问题, 本文将直觉模糊集相似度与GA相结合, 用以求解. 根据置换流水车间调度问题的描述可知, 工件的加工路径和所有机器上工件的加工次序相同, 因此本文采用整数对调度工序进行编码, 将每个染色体对应转化为*n*个工件的一个排列. 如一条染色体 $x = \{x_{i1}, x_{i2}, \dots, x_{in}\}$, 其中*i*表示工件号($1 < i < n + 1$), 每个*i*

出现次数为1, 使染色体每个位置分量对应为工件排列中的一个工件号. 通过ROV(rank order value)^[15]的方法实现GA算法连续的实数到离散工序的转化. 算法的具体步骤如下:

Step 1 产生初始种群.

初始化参数, 随机生成 NP 个基因 X_j^{gen} 形成初始父代种群, 其中gen为种群当前进化的代数. 建立初始外部档案.

Step 2 建立参考解直觉模糊集.

利用GA算法分别用式(1)–(4)进行优化求解, 构造参考解函数值 Y_0 . 将 Y_0 映射为参考解直觉解模糊集IFS(0).

Step 3 建立Pareto解直觉模糊集.

计算各基因的目标函数值, 得到Pareto前沿 Y_j^{gen} , 将 Y_j^{gen} 映射为Pareto解直觉模糊集IFS^{gen}(j).

Step 4 计算直觉模糊集之间的相似度.

按照式(19)计算第gen代的每个基因 X_j^{gen} 的IFS^{gen}(j)与IFS(0)的相似度 $S(\text{IFS}(j), \text{IFS}(0))$, 并将其作为GA算法的适应度值来引导算法进化.

Step 5 选择、交叉和变异操作.

采用二元锦标赛^[16]方法选择较优的基因, 采用部分映射交叉(PMX)^[17]方法及互换变异(SWAP)^[16]方法产生新的基因(Pareto解), 形成新的子代种群, 其中 P_c 为交叉概率, P_m 为变异概率.

Step 6 外部档案维护和更新.

合并子代种群和外部档案, 计算其拥挤距离, 将具有较大拥挤距离的Pareto解保留, 删除拥挤距离小的Pareto解对其进行修剪, 从而更新的外部档案.

Step 7 判断是否满足终止条件.

判断是否满足群体适应度值连续一定次数不发生变化或者是否满足设定的最大代数maxgen, 不满足则gen = gen + 1, 跳入Step3; 否则, 结束迭代.

4 仿真研究

为了验证SIFS_GA算法的优越性以及求解多目标置换流水车间调度问题的可行性及有效性, 本文分别选取6个最新CEC标准测试集^[17], 以及10个不同规模的流水车间测试实例^[18], 来进行仿真测试. 针对CEC标准测试集, 选取NSGAIII^[19], NSGAII^[20], g_GA^[21]、(g-dominance genetic algorithm)以及SPEA2^[22]作为对比算法; 针对流水车间调度问题的测试实验中, 选取NSGAII, RW_GA(random weight genetic algorithm)以及DABC(discrete artificial bee colony algorithm)^[23]作为对比算法.

NSGAII是改进非支配排序遗传算法, 它是一种经典的多目标优化算法, 目前已有很多学者将其应用到

多目标流水车间调度中; NSGAIII是近年来提出的基于参考点的非支配排序的遗传算法, 是一种较新的多目标优化算法. g_GA是基于g-支配的遗传算法, 是一种常用的多目标优化算法; SPEA2是改进强度帕累托进化算法, 是一种经典的多目标优化算法; RW_GA是基于随机权重的遗传算法, DABC是离散的人工蜂群算法, 后两种算法均已被应用于求解多目标流水车间调度问题.

5 实验参数

为了更好的进行比较, 算法共同的参数设置如下: 种群大小 $NP = 20$, 外部档案大小 $W_{\text{max}} = 20$, 最大进化代数 $\text{maxgen} = 100$. GA交叉概率 $P_c = 0.9$, 变异概率 $P_m = 0.1$, 采用遗传算法进行每一个子目标的单目标优化次数 $q = 10$. 在SIFS_GA算法中, 定界系数 $\lambda = 0.9$, 非隶属度函数的参数 $\alpha_i = 1.2$ ^[10]; 在DABC算法中雇佣蜂和跟随蜂数量都是种群大小的50%.

CEC标准测试集中测试函数的参数设置如下: 选取MaF1, MaF2, MaF4, MaF5, MaF10, MaF11这6个标准测试函数, 设置函数的目标个数 $M = 4$, 其他的参数设置情况见参考文献[17], 标准测试函数的代码可在网站 (<http://www.cercia.ac.uk/news/cec2017maoc/>) 下载.

流水车间测试实例的参数设置如下: 完工工件每批最大运输量 $h = 5$, 单位库存成本 $\beta_i = 1$, 单位拖期成本 $\gamma_i = 2$.

将SIFS_GA的参考解作为g_GA和NSGAIII的参考点. 每个测试样例和测试实例都用不同的随机种子优化10次. 算法中的一些重要参数设置是通过研究相关文献和大量实验来确定的.

5.1 算法性能评价指标

为了验证算法的性能, 本文采用3个性能评价指标: 间隔距离(SP)、当代距离(GD)以及C指标.

1) 多样性指标. 间隔距离(SP)^[24], 用以评价解集在目标空间上分布均匀性. 计算公式如下:

$$SP = \left(\sum_{i=1}^{W_{\text{max}}} (\bar{d} - d_i)^2 / (W_{\text{max}} - 1) \right)^{1/2}, \quad (20)$$

$$d_i = \min_j (|f_1^i - f_1^j| + |f_2^i - f_2^j| + |f_3^i - f_3^j| + |f_4^i - f_4^j|),$$

式中: $i, j = 1, \dots, W_{\text{max}}$; \bar{d} 为所有 d_i 的均值. SP值越小, 解集分布越均匀.

2) 收敛性指标. 当代距离(GD)^[25], 用以评价算法的收敛性. 公式如下:

$$GD = \left(\sum_{i=1}^{W_{\text{max}}} d_i^2 \right)^{1/2} / W_{\text{max}}, \quad (21)$$

式中: d_i 为第*i*个非支配解与参考解之间的欧氏距离. GD值越小, Pareto解越逼近参考解, 收敛性越好.

3) 覆盖率指标. C指标^[26], 用以度量两种算法 \bar{A} , \bar{B} 优化同一问题得到的两个最优解集*A*, *B*之间的相对优劣程度, 计算公式如下:

$$C(\bar{A}, \bar{B}) = \frac{|\{b \in B | \exists a \in A, \text{ s.t. } a \prec b\}|}{|B|}, \quad (22)$$

式中: $C(\bar{A}, \bar{B})$ 的值表示*A*中的解非劣于*B*中的解的百分比, $C(\bar{A}, \bar{B}) \in [0, 1]$. $C(\bar{A}, \bar{B})$ 值越大表明解集*A*越优于解集*B*. 算法 \bar{A} 越优于算法 \bar{B} .

5.2 结果比较与分析

5.2.1 CEC测试集测试结果

从CEC测试集中选取的6个MaF标准测试函数具有不同的特性, 它们的理论前沿面具有不同的形状, 能够反映实际优化问题具有的复杂性质, 因此, 可以

检测所提进化高维多目标优化方法解决实际优化问题的性能.

各种算法的评价指标平均值如表1所示. 在间隔距离(SP)方面, 在实例MaF2, MaF5, MaF10中, SIFS_GA的SP值均小于其他4个算法; 在实例MaF1, MaF4中, SIFS_GA的SP值大于SPEA2, 但是小于其他3个算法; 在实例MaF11中, SIFS_GA的SP值大于NSGAIII与g_GA的值, 但是小于NSGAI, SPEA2的值; 表明整体上来说, 算法SIFS_GA求得的Pareto解集分布的均匀性优于其他算法. 在当代距离(GD)方面, 在实例MaF1, MaF2, MaF4, MaF5, MaF11中, SIFS_GA的GD值均小于其他4个算法; 在实例MaF10中, SIFS_GA的GD值大于NSGAIII, 但是小于其他3个算法; 表明SIFS_GA求得的Pareto解更逼近参考解, 其收敛性比NSGAIII, NSGAI, g_GA以及SPEA2好.

表 1 各种算法求解6个CEC标准测试集的性能评价指标结果(指标最好值用加粗强调)

Table 1 Performance evaluation index results of various CEC standard test sets by various algorithms (the best value of the indicator uses bold emphasis)

| 测试样例 | 算法 | SP | GD | 测试样例 | 算法 | SP | GD |
|------|---------|----------------|----------------|-------|---------|--------------|---------------|
| MaF1 | SIFS_GA | 0.201 | 0.396 | MaF5 | SIFS_GA | 0.157 | 2.425 |
| | NSGAIII | 4.077 | 12.346 | | NSGAIII | 162.629 | 164.322 |
| | NSGAI | 0.314 | 0.418 | | NSGAI | 0.948 | 2.792 |
| | g_GA | 0.347 | 0.598 | | g_GA | 1.817 | 4.653 |
| | SPEA2 | 0.105 | 0.49 | | SPEA2 | 0.964 | 2.428 |
| MaF2 | SIFS_GA | 0.041 | 0.13 | MaF10 | SIFS_GA | 0.008 | 0.069 |
| | NSGAIII | 1.341 | 4.537 | | NSGAIII | 0.015 | 0.029 |
| | NSGAI | 0.133 | 0.156 | | NSGAI | 0.081 | 0.097 |
| | g_GA | 0.103 | 0.153 | | g_GA | 0.042 | 0.098 |
| | SPEA2 | 0.055 | 0.167 | | SPEA2 | 0.029 | 0.091 |
| MaF4 | SIFS_GA | 674.534 | 553.519 | MaF11 | SIFS_GA | 1.346 | 10.112 |
| | NSGAIII | 1079.272 | 1177.895 | | NSGAIII | 0.061 | 15.831 |
| | NSGAI | 1880.519 | 1536.621 | | NSGAI | 4.446 | 11.13 |
| | g_GA | 1752.776 | 2320.991 | | g_GA | 0.776 | 16.124 |
| | SPEA2 | 459.293 | 1602.694 | | SPEA2 | 1.933 | 12.006 |

5.2.2 流水车间测试实例测试结果

参考文献[18]的方法建立10个不同规模的测试实例. 测试实例的规模从10个工件数、5个机器数到100个工件数、20个机器数.

仿真结果为10次运行的平均值, 如表2所示. 在实例2中, SIFS_GA有4个目标优于NSGAI与RW_GA, 有2个目标优于DABC; 在实例1, 3, 4, 5, 6, 7, 8, 9, 10中, SIFS_GA所得多目标最优解均优于其他3个算法; 这表明在求解多目标置换流水车间调度的问题上, SIFS_GA得到的多目标优化解优于NSGAI, RW_GA

和DABC三个算法. 根据表2可知10个实例的直觉模糊集相似度(S)均大于0.8, 表明SIFS_GA所得Pareto最优解与参考解比较相似, 以直觉模糊集相似度引导算法进化的效果较优.

各种算法的评价指标结果如表3所示. 在间隔距离(SP)方面, SIFS_GA的SP值均小于其他3个算法, 表明算法SIFS_GA求得的Pareto解集分布的均匀性优于其他算法. 在当代距离(GD)方面, 在所有实例中, SIFS_GA的GD值均小于其他3个算法, 表明SIFS_GA求得的Pareto解更逼近参考解, 其收敛性是最优的.

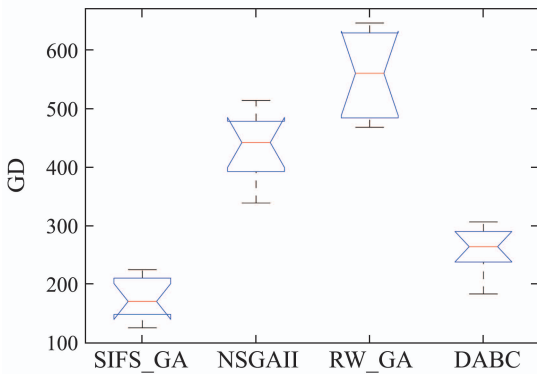
表 2 10个置换流水车间测试实例的仿真实验结果(最好的优化解用加粗强调)

Table 2 Simulation results of 10 permutation flow shop test cases (the best optimization solution uses bold emphasis)

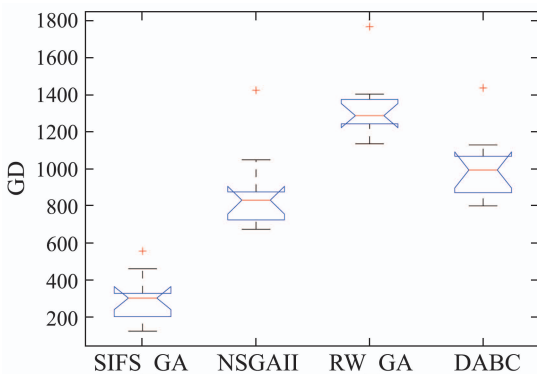
| 实例 序号 | 算法 | 问题 ($n \times m$) | 参考解 $Y_0 = (f_1, f_2, f_3, f_4, f_5)$ | 多目标优化解 $(f_1, f_2, f_3, f_4, f_5)$ | 相似度S |
|----------|---------|------------------------|--|---|------|
| 1 | SIFS_GA | 10×5 | (835, 245, 964, 1585) | (878, 301, 1133, 2027) | 0.88 |
| | NSGAI | | | (908, 571, 1309, 3030) | |
| | RW_GA | | | (999, 542, 1766, 3676) | |
| | DABC | | | (910, 313, 1420, 2052) | |
| 2 | SIFS_GA | 10×10 | (1036, 183, 838, 1132) | (1050, 233, 979, 1371) | 0.92 |
| | NSGAI | | | (1134, 356, 1138, 2162) | |
| | RW_GA | | | (1230, 502, 1711, 3364) | |
| | DABC | | | (1122, 207 , 1244, 1299) | |
| 3 | SIFS_GA | 20×5 | (1296, 380, 1678, 1888) | (1323, 462, 1826, 2298) | 0.92 |
| | NSGAI | | | (1365, 791, 2091, 4485) | |
| | RW_GA | | | (1511, 840, 2678, 4436) | |
| | DABC | | | (1428, 466, 2398, 2654) | |
| 4 | SIFS_GA | 20×10 | (1812, 805, 2122, 6904) | (1880, 979, 2283, 7716) | 0.88 |
| | NSGAI | | | (1942, 1302, 2726, 10060) | |
| | RW_GA | | | (2041, 1487, 3328, 10837) | |
| | DABC | | | (1975, 1104, 2725, 10345) | |
| 5 | SIFS_GA | 40×10 | (3030, 2158, 4410, 55231) | (3036, 2281, 4703, 56987) | 0.9 |
| | NSGAI | | | (3138, 2460, 5155, 61649) | |
| | RW_GA | | | (3372, 3015, 6260, 70943) | |
| | DABC | | | (3232, 2684, 5483, 67922) | |
| 6 | SIFS_GA | 40×20 | (3805, 2496, 4729, 39029) | (3855, 2598, 5055, 41800) | 0.89 |
| | NSGAI | | | (4025, 2917, 5798, 47696) | |
| | RW_GA | | | (4168, 3450, 6868, 52750) | |
| | DABC | | | (4067, 3150, 5868, 53798) | |
| 7 | SIFS_GA | 50×20 | (4439, 3164, 6048, 81178) | (4504, 3339, 6208, 85398) | 0.89 |
| | NSGAI | | | (4612, 3514, 7043, 90562) | |
| | RW_GA | | | (4880, 4328, 8312, 100202) | |
| | DABC | | | (4751, 3729, 7552, 104394) | |
| 8 | SIFS_GA | 60×20 | (5181, 3857, 7342, 103623) | (5029, 3963, 7571, 106259) | 0.89 |
| | NSGAI | | | (5315, 4296, 8249, 114985) | |
| | RW_GA | | | (5539, 4888, 9331, 132431) | |
| | DABC | | | (5483, 4595, 8866, 121752) | |
| 9 | SIFS_GA | 80×20 | (6347, 4198, 9544, 104914) | (6403, 4538, 9698, 106572) | 0.89 |
| | NSGAI | | | (6527, 4908, 10657, 124889) | |
| | RW_GA | | | (6743, 5538, 12056, 129987) | |
| | DABC | | | (6637, 4915, 11231, 126056) | |
| 10 | SIFS_GA | 100×20 | (7472, 5624, 11711, 206859) | (7519, 5869, 11913, 208313) | 0.89 |
| | NSGAI | | | (7712, 6174, 13113, 217208) | |
| | RW_GA | | | (7980, 6967, 14595, 247182) | |
| | DABC | | | (7828, 6374, 13756, 240925) | |

为了更加直观的比较每种算法的性能,4种算法的GD值的箱线图如图1所示. 因为篇幅所限, 仅列出6个实例的结果. 每个箱线图由10组仿真数据组成. 其中箱线图中间的红线代表中位数, 红色‘+’是异常值; 箱线图越窄意味着方差越小, 稳定性越高. 在大多数实例中, SIFS_GA对应的箱线图的宽度最窄, 表示稳定性好. GD的中位数越小, 算法越接近参考解, 收敛性越好. SIFS_GA的中线低于其他3种算法, 说明其解集的质量较好.

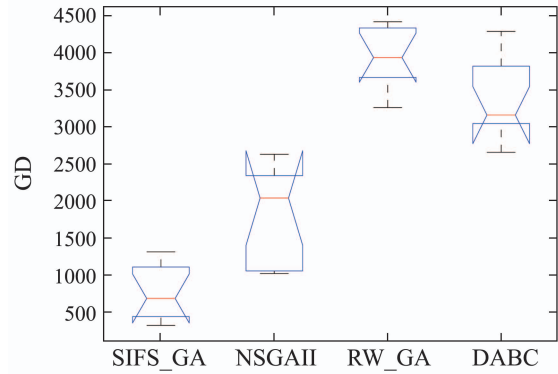
各算法覆盖率指标比较如表4所示. 从表中可以看出, 对于RW_GA, $C(SIFS_GA, RW_GA)$ 的值均接近1, 而 $C(RW_GA, SIFS_GA)$ 的值均接近0, 表明SIFS_GA的优化效果明显优于RW_GA; 对于NSGAI, $C(SIFS_GA, NSGAI)$ 的值均接近1且远大于 $C(NSGAI, SIFS_GA)$, 表明SIFS_GA获得的大多数解覆盖了NSGAI获得的解, 而NSGAI获得的解只有少一部分覆盖了SIFS_GA获得的解; 对于DABC, $C(SIFS_GA, DABC)$ 的值均大于 $C(DABC, SIFS_GA)$ 的值, 表明SIFS_GA获得的解覆盖DABC获得解的比例大于DABC获得的解覆盖SIFS_GA获得解的比例. 另一方面, 随着规模的增大, 各个算法与SIFS_GA相对应的C指标的差值有逐渐增大的趋势, 表明在解决大规模问题时SIFS_GA得到的解更优. 得出结论, SIFS_GA在求解时多目标流水线车间调度问题时优于NSGAI, RW_GA和DABC, 并且更适合求解大规模问题.



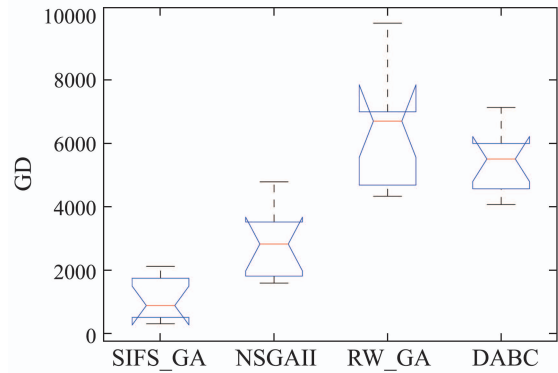
(a) 10×5



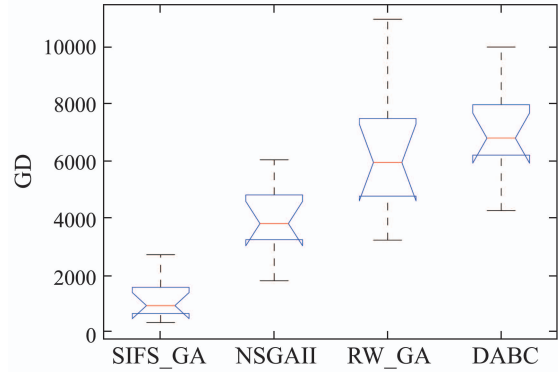
(b) 20×10



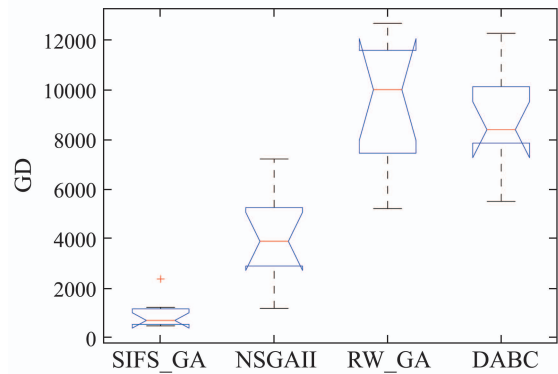
(c) 40×10



(d) 60×20



(e) 80×20



(f) 100×20

图1 GD值箱线图

Fig. 1 Value of GD box plot

表 3 各种算法求解10个流水车间测试实例的性能评价指标结果

Table 3 Performance evaluation index results of flow shop test examples by various algorithms

| 实例序号 | 算法 | 问题 | SP | GD | 实例序号 | 算法 | 问题 | SP | GD |
|------|---------|---------|---------------|---------------|------|---------|----------|---------------|----------------|
| 1 | SIFS_GA | 10 × 5 | 59.49 | 176.14 | 6 | SIFS_GA | 40 × 20 | 242.94 | 911.99 |
| | NSGAI | | 180.44 | 434.64 | | NSGAI | | 862.61 | 2024.33 |
| | RW_GA | | 218.31 | 554 | | RW_GA | | 759.12 | 3592.82 |
| | DABC | | 213.08 | 255.85 | | DABC | | 1290.16 | 3680.29 |
| 2 | SIFS_GA | 10 × 10 | 65.51 | 105.47 | 7 | SIFS_GA | 50 × 20 | 492.05 | 1259.2 |
| | NSGAI | | 247.89 | 275.06 | | NSGAI | | 1761.16 | 2696.49 |
| | RW_GA | | 179.29 | 651.44 | | RW_GA | | 1520.19 | 4515.54 |
| | DABC | | 210.68 | 176.17 | | DABC | | 1665.2 | 5171.58 |
| 3 | SIFS_GA | 20 × 5 | 99.76 | 189.42 | 8 | SIFS_GA | 60 × 20 | 307.71 | 1061.56 |
| | NSGAI | | 254.38 | 541.07 | | NSGAI | | 1081.8 | 2817.85 |
| | RW_GA | | 307.92 | 1066.7 | | RW_GA | | 1337.16 | 6420.53 |
| | DABC | | 379.71 | 371.32 | | DABC | | 1348.09 | 5485.89 |
| 4 | SIFS_GA | 20 × 10 | 117.1 | 304.76 | 9 | SIFS_GA | 80 × 20 | 435.83 | 1180.37 |
| | NSGAI | | 445.6 | 877.3 | | NSGAI | | 1508.65 | 3946.04 |
| | RW_GA | | 441.49 | 1301.65 | | RW_GA | | 2686.36 | 6367.67 |
| | DABC | | 487.51 | 1011.32 | | DABC | | 2721.36 | 7123.16 |
| 5 | SIFS_GA | 40 × 10 | 293.35 | 750.37 | 10 | SIFS_GA | 100 × 20 | 586.29 | 903.77 |
| | NSGAI | | 672.86 | 1856.34 | | NSGAI | | 2885.58 | 4103.28 |
| | RW_GA | | 685.17 | 3913.91 | | RW_GA | | 1691.45 | 9507.96 |
| | DABC | | 1167.77 | 3349.07 | | DABC | | 3558.38 | 8706.28 |

表 4 各算法覆盖率指标比较

Table 4 Comparison of coverage indicators of each algorithm

| 实例序号 | 问题 (n × m) | C(SIFS_GA, NSGAI) | C(NSGAI, SIFS_GA) | C(SIFS_GA, RW_GA) | C(RW_GA, SIFS_GA) | C(SIFS_GA, DABC) | C(DABC, SIFS_GA) |
|------|------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|
| 1 | 10 × 5 | 1 | 0.41 | 1 | 0 | 1 | 0.575 |
| 2 | 10 × 10 | 1 | 0.34 | 1 | 0 | 1 | 0.455 |
| 3 | 20 × 5 | 1 | 0.32 | 1 | 0 | 1 | 0.275 |
| 4 | 20 × 10 | 1 | 0.33 | 1 | 0.005 | 1 | 0.18 |
| 5 | 40 × 10 | 1 | 0.385 | 1 | 0 | 1 | 0.075 |
| 6 | 40 × 20 | 1 | 0.215 | 1 | 0 | 1 | 0.045 |
| 7 | 50 × 20 | 1 | 0.205 | 1 | 0 | 1 | 0.045 |
| 8 | 60 × 20 | 1 | 0.2 | 1 | 0 | 1 | 0.02 |
| 9 | 80 × 20 | 1 | 0.255 | 1 | 0 | 1 | 0.025 |
| 10 | 100 × 20 | 1 | 0.265 | 1 | 0 | 1 | 0.01 |

6 结论

本文针对高维多目标置换流水车间调度问题, 设计了基于直觉模糊集相似度的方法求解. 将参考解和 Pareto 解分别映射为直觉模糊集, 计算 Pareto 解直觉模糊集和参考解直觉模糊集的相似度并以其作为进化算法的适应度值, 建立基于直觉模糊集相似度的遗传算法(SIFS_GA), 用以求解4个目标的置换流水车间调度问题. 通过仿真结果表明, 直觉模糊集相似度可以

和遗传算法有效结合, 算法性能优于常用的多目标优化算法, 且在多目标置换流水车间调度问题上可以获得较高质量的 Pareto 解, 更适用于求解大规模的问题. 本文在求解车间调度问题时选择了置换流水车间调度问题, 实际生产中可能出现更为复杂的情况, 如零空闲流水车间调度、阻塞流水车间调度、作业车间调度^[27]、柔性作业车间调度^[28]、动态调度等. 下一步将通过改进算法, 探索上述问题, 进一步扩展本文

所提算法的应用范围.

参考文献:

- [1] PAN Quanke, GAO Liang, LI Xinyu. *Flow Shop Scheduling and Its Optimization Algorithm*. Wuhan: Huazhong University of Science and Technology Press, 2013: 20 – 21.
(潘全科, 高亮, 李新宇. 流水车间调度及其优化算法. 武汉: 华中科技大学出版社, 2013: 20 – 21.)
- [2] WANG Dazhi, LIU Shixin, GUO Xiwang. A multi-agent evolutionary algorithm for solving total tardiness permutation flow-shop scheduling problem. *Acta Automatica Sinica*, 2014, 40(3): 548 – 555.
(王大志, 刘士新, 郭希旺. 求解总拖期时间最小化流水车间调度问题的多智能体进化算法. 自动化学报, 2014, 40(3): 548 – 555.)
- [3] ZHANG Xianchao, ZHOU Hong. Variable parameters quantum-inspired evolutionary algorithm and its application in permutation flow-shop scheduling problem. *Computer Integrated Manufacturing System*, 2016, 22(3): 774 – 781.
(张先超, 周泓. 变参数量子进化算法及其在求解置换流水车间调度问题中的应用. 计算机集成制造系统, 2016, 22(3): 774 – 781.)
- [4] LIN J. A hybrid discrete biogeography-based optimization for the permutation flow shop scheduling problem. *International Journal of Production Research*, 2017, 54(16): 1 – 10.
- [5] LIU Gang, HUANG Chongzheng. Permutation flow shop scheduling based on hybrid discrete artificial bee colony algorithm. *Control Engineering of China*, 2017, 24(9): 1925 – 1929.
(刘刚, 黄崇争. 基于HDABC算法的置换流水车间调度策略. 控制工程, 2017, 24(9): 1925 – 1929.)
- [6] LU C, GAO L, LI X, et al. Energy-efficient permutation flow shop scheduling problem using a hybrid multi-objective backtracking search algorithm. *Journal of Cleaner Production*, 2017, 144: 228 – 238.
- [7] FU Yaping, HUANG Min, WANG Hongfeng, et al. Multipopulation multiobjective genetic algorithm for multiobjective permutation flow shop scheduling problem. *Control Theory & Applications*, 2016, 33(10): 1281 – 1288.
(付亚平, 黄敏, 王洪峰, 等. 面向多目标流水车间调度的多种群多目标遗传算法. 控制理论与应用, 2016, 33(10): 1281 – 1288.)
- [8] NGUYEN H. A novel similarity/dissimilarity measure for intuitionistic fuzzy sets and its application in pattern recognition. *Expert Systems with Applications*, 2016, 45(C): 97 – 107.
- [9] ANANTHI V P, BALASUBRAMANIAM P, RAVEENDRAN P. A thresholding method based on interval-valued intuitionistic fuzzy sets: an application to image segmentation. *Pattern Analysis & Applications*, 2018, 21(4): 1039 – 1051.
- [10] YANG J, LI J, WANG Y, et al. Optimization of weapon-target assignment problem by intuitionistic fuzzy genetic algorithm. *Application Research of Computers*, 2018, DOI: 10.1051/mateconf/201712802004.
- [11] WANG Kaijun, LIN Pinle. Operational safety and reliability assessment of high speed train with intuitionistic fuzzy set and VIKOR method. *Computer Systems & Applications*, 2017, 26(4): 29 – 34.
(王开军, 林品乐. 基于直觉模糊集的灰色模型故障预测. 计算机系统应用, 2017, 26(4): 29 – 34.)
- [12] LEI Deming, YAN Xiping. *Multi-objective Intelligent Optimization Algorithm and Its Application*. Beijing: Science Press, 2009: 271 – 272.
(雷德明, 严新平. 多目标智能优化算法及其应用. 北京: 科学出版社, 2009: 271 – 272.)
- [13] ATANASSOV K. Intuitionistic fuzzy sets. *Fuzzy Sets and System*, 1986, 20(1): 87 – 96.
- [14] HE Zhenghong. *Research on intuitionistic fuzzy clustering and its application in information fusion*. XI'an: Air Force Engineering University, 2010.
(贺正洪. 直觉模糊聚类及其在信息融合中的应用研究. 西安: 空军工程大学, 2010.)
- [15] TAO Q, CHANG H, YI Y, et al. A rotary chaotic PSO algorithm for trustworthy scheduling of a grid workflow. *Computers & operations research*, 2011, 38(5): 824 – 836.
- [16] WANG Ling. *Shop Scheduling with Genetic Algorithms* [M]. Beijing: Qinghua University Press, 2003: 51 – 53.
(王凌. 车间调度遗传算法. 北京: 清华大学出版社, 2003: 51 – 53.)
- [17] CHENG R, LI M, TIAN Y, et al. A benchmark test suite for evolutionary many-objective optimization. *Complex & Intelligent Systems*, 2017, 3(1): 67 – 81.
- [18] ISHIBUCHI H, YOSHIDA T, MURATA T. Balance between genetic search and local search in memetic algorithms for multi-objective permutation flows shop scheduling. *IEEE Transactions on Evolutionary Computation*, 2003, 7(2): 204 – 223.
- [19] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 2014, 18(4): 577 – 601.
- [20] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182 – 197.
- [21] MOLINAC J, HERNÁNDEZ-DÍAZ A G, COELLO C A C, et al. g-dominance: reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, 2009, 197(2): 685 – 692.
- [22] LAUMANN M. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*. Switzerland: Technical Report Gloriastrasse, 2001, 103.
- [23] PAN Q K, TASGETIREN M F, SUGANTHAN P N, et al. A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Computer Engineering & Applications*, 2011, 181(12): 2455 – 2468.
- [24] GONG M, JIAO L, DU H, et al. Multiobjective immune algorithm with nondominated neighbor-based selection. *Evolutionary Computation*, 2009, 17(1): 131.
- [25] ABRAHAM A, JAIN L. *Evolutionary multi-objective optimization. Theoretical Advances and Applications*. Beilin, Germany: Springer-Verlag, 2006: 18 – 20.
- [26] ZITZLER E, THIELE L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *Evolutionary Computation*, 1999, 3(4): 257 – 271.
- [27] CAO Yangming, ZHOU Yaqin, YANG Jianguo, et al. The research on job-shop scheduling by considering tool constraints. *Modern Manufacturing Engineering*, 2017, (12): 114 – 118.
(曹阳明, 周亚勤, 杨建国, 等. 考虑刀具约束的作业车间调度研究. 现代制造工程, 2017, (12): 114 – 118.)
- [28] AI Ziyi, LEI Deming. A novel shuffled frog leaping algorithm for low carbon flexible job shop scheduling. *Control Theory & Applications*, 2017, 34(10): 1361 – 1368.
(艾子义, 雷德明. 基于新型蛙跳算法的低碳柔性作业车间调度. 控制理论与应用, 2017, 34(10): 1361 – 1368.)

作者简介:

徐文婕 硕士研究生, 主要研究方向为智能优化调度, E-mail: 861

763958@qq.com;

朱光宇 博士, 教授, 主要研究方向为智能制造等, E-mail: zhugy

@fzu.edu.cn.