

控制与决策

Control and Decision



离散猫群优化算法求解带交货期的FJSP问题

姜天华, 邓冠龙, 朱惠琦

引用本文:

姜天华, 邓冠龙, 朱惠琦. 离散猫群优化算法求解带交货期的FJSP问题[J]. 控制与决策, 2020, 35(1): 161–168.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2018.0631>

您可能感兴趣的其他文章

Articles you may be interested in

基于动态参数的人工搜索群算法

Artificial search swarm algorithm based on dynamic parameters

控制与决策. 2019, 34(9): 1923–1928 <https://doi.org/10.13195/j.kzyjc.2018.0099>

考虑能耗与质量的机床构件生产线多目标柔性作业车间调度方法

Multi-objective flexible job shop scheduling method for machine tool component production line considering energy consumption and quality

控制与决策. 2019, 34(2): 252–260 <https://doi.org/10.13195/j.kzyjc.2018.0131>

差异容量平行批机器环境下基于弱选择约束的调度算法

Weak-restriction based algorithm for scheduling on parallel batch machines with arbitrary capacities

控制与决策. 2018, 33(8): 1363–1372 <https://doi.org/10.13195/j.kzyjc.2017.0524>

混沌海豚群优化灰色神经网络的空中目标威胁评估

Air-targets threat assessment using grey neural network optimized by chaotic dolphin swarm algorithm

控制与决策. 2018, 33(11): 1997–2003 <https://doi.org/10.13195/j.kzyjc.2017.0812>

传摘时间约束下的运输与批处理机生产协调调度

Coordinated scheduling of transportation and production on batching-machine with tracking time constraint

控制与决策. 2017, 32(6): 995–1000 <https://doi.org/10.13195/j.kzyjc.2016.0489>

协调探索和开发能力的改进灰狼优化算法

Improved grey wolf optimization algorithm coordinating the ability of exploration and exploitation

控制与决策. 2017, 32(10): 1749–1757 <https://doi.org/10.13195/j.kzyjc.2016.1545>

应急救援物资多目标分配与调度问题建模与求解

Modeling and solving multi-objective allocation-scheduling of emergency relief supplies

控制与决策. 2017, 32(1): 86–92 <https://doi.org/10.13195/j.kzyjc.2015.1518>

改进型细菌觅食算法及多货叉仓库调度应用

Improved bacterial foraging algorithm and application of warehousing scheduling with multi-shuttles

控制与决策. 2015(2): 321–327 <https://doi.org/10.13195/j.kzyjc.2013.1436>

离散猫群优化算法求解带交货期的FJSP问题

姜天华^{1†}, 邓冠龙², 朱惠琦¹

(1. 鲁东大学 交通学院, 山东 烟台 264025; 2. 鲁东大学 信息与电气工程学院, 山东 烟台 264025)

摘要: 针对带交货期的柔性作业车间调度问题(flexible job shop scheduling problem, FJSP), 提出一种离散猫群优化算法(discrete cat swarm optimization, DCSO), 以优化工件最大完工时间和平均提前/拖期时间. 首先, 设计一种两段式离散编码方式, 用于表示调度解, 并采用启发式算法实现种群初始化; 其次, 为了使算法能够直接在离散调度空间内运行, 在搜寻模式下设计基于 3 种不同邻域结构的搜寻方法, 并在跟踪模式下提出一种新型离散个体更新公式; 再次, 采用线性自适应猫群行为模式选择策略, 协调算法全局搜索和局部搜索的能力; 最后, 为了进一步改善计算结果, 在算法中嵌入一种局部搜索策略. 通过基准算例测试 DCSO 算法的性能, 仿真结果表明所提 DCSO 算法在求解 FJSP 问题方面的有效性.

关键词: 柔性作业车间; 最大完工时间; 平均提前/拖期时间; 离散猫群优化算法

中图分类号: TH165

文献标志码: A

Discrete cat swarm optimization algorithm for solving the FJSP with due date

JIANG Tian-hua^{1†}, DENG Guan-long², ZHU Hui-qi¹

(1. School of Transportation, Ludong University, Yantai 264025, China; 2. School of Information and Electrical Engineering, Ludong University, Yantai 264025, China)

Abstract: For the flexible job shop scheduling problem (FJSP) with due date, a discrete cat swarm optimization (DCSO) algorithm is proposed to optimize the makespan and the mean earliness/tardiness. Firstly, a two-phase discrete encoding approach is designed to represent the scheduling solution, and a heuristic-based method is adopted to fulfill the population initialization. Then, in order to make the algorithm work directly in a discrete scheduling space, a seeking method is developed based on three different neighborhood structures in the seeking mode, and a new discrete individual updating equation is proposed in the tracking mode. To coordinate the global and local search capability of the proposed algorithm, a linear adaptive selection strategy of cat's behavior modes is used. In addition, a local search strategy is embedded into the algorithm to further improve the computational results. Finally, the performance of the DCSO algorithm is tested based on the benchmark instances, and experimental data demonstrate the effectiveness of the proposed DCSO algorithm on solving the FJSP.

Keywords: flexible job shop; makespan; mean earliness/tardiness; discrete cat swarm optimization

0 引言

柔性作业车间调度问题作为经典作业车间调度问题的拓展形式, 具有很强的工业应用背景. 该类问题在作业车间调度问题的基础上增加了对调度灵活性的考虑, 使工件具有柔性的加工路径, 因此被认为是最困难的组合优化问题之一. 一般柔性作业车间调度问题可分为两个子问题: 机器分配问题和工序排序问题. 即在工序加工之前, 首先为工序分配机器, 然后再排列机器上工序间的加工顺序, 以使车间预期

生产指标能够达到最优.

目前有关柔性作业车间调度问题的研究有很多, 优化目标一般为最大完工时间、最大机器负载、机器总负载以及与交货期相关的指标等. 虽然许多文献以工件最大完工时间作为优化目标, 但在实际生产中, 尤其是准时制生产环境下, 与交货期相关的性能指标也是必须考虑在内的, 如提前期时间/成本、拖期时间/成本以及二者的组合等^[1]. Gao 等^[1]提出了一种离散和声搜索算法, 用于优化柔性作业车间内工件最

收稿日期: 2018-05-12; 修回日期: 2018-08-02.

基金项目: 国家自然科学基金青年基金项目(61403180); 山东省自然科学基金项目(ZR2016GP02); 山东省自然科学基金项目(ZR2019QF008); 山东省高等学校科技计划项目(J17KA199).

[†]通讯作者. E-mail: jth1127@163.com.

大完工时间和平均提前期/拖期时间; Mousakhani^[2]针对生产准备时间与工序排序相关的柔性作业车间调度问题,以工件总拖期时间最小为优化目标,建立了一种混合整数线性规划模型,并基于迭代局部搜索算法提出了一种元启发式算法进行求解; Scrich等^[3]在禁忌搜索算法的基础上提出了两种启发式算法,用于求解柔性作业车间内的总拖期时间; Wu等^[4]提出了一种基于多智能体的调度方法,用于求解柔性作业车间内的提前期和拖期时间加权和; Türkyılmaz等^[5]以总拖期时间为优化目标,将遗传算法和并行变邻域搜索算法相结合,提出了一种混合算法对柔性作业车间调度问题进行求解; Na等^[6]针对柔性作业车间内工件具有层次结构的特点,建立了以总拖期时间最小为目标的数学模型,并提出了一种遗传算法对其进行求解; Huang等^[7]提出了一种双信息素蚁群算法,用于优化柔性作业车间的总提前期和总拖期时间的加权和; 刘晓霞等^[8]以生产周期、总拖期时间和机器负载为优化目标,设计了一种基于 Pareto 最优的混合遗传算法; 王云等^[9]以制造工期、加工成本和提前期/拖期惩罚为优化目标,提出了一种改进的 Pareto 进化算法; Vilcot等^[10]提出了两种禁忌搜索算法,以优化实际柔性作业车间中工件最大完工时间和最大拖期时间。

柔性作业车间调度问题已被证明具有很强的 NP 难特性,这使得智能优化算法成为求解该类问题的一种有效且最受欢迎的途径。目前,各种智能优化算法均存在其优缺点,仍没有一种算法能够获得所有车间调度问题的最优解,这也使得智能算法的研究成为车间调度领域的研究热点之一。猫群优化算法(cat swarm optimization, CSO)是基于猫的日常工作习性而提出的一种群智能优化算法,该算法区别于其他算法的主要特点在于每次迭代时都将猫群按比例随机划分为两个子群,并分别在搜寻模式和跟踪模式下进行个体位置更新,使算法能够同时进行全局和局部搜索,有利于实现算法全局和局部搜索能力的平衡^[11]。该算法目前已被应用于求解多种复杂优化问题,如函数优化^[12]、图像处理^[13]、特征选择与参数优化^[14]、排课问题^[15]等,但在车间调度领域的研究相对较少。文献[16]提出了一种多目标猫群优化算法,用于优化混流装配线工作站闲置/超载总成本、产品变化率和产品切换总时间。文献[17]提出了一种改进型猫群优化算法,用于优化柔性作业车间内工件的最大完工时间。基本猫群优化算法最初是为求解连续优化问题而提出的,猫群个体位置在连续域内进行更新,然而车间调度问题中每个调度解却具有离散的特

性,这导致基本猫群优化算法无法直接用于车间调度问题。对此,文献[16]和文献[17]均提出了一种能够实现连续个体位置与离散调度解间的转换方法,但该方法需要二者在进化过程中频繁地进行转换,这必然会额外增加算法的计算时间。因此,针对带交货期的柔性作业车间调度问题的特点,本文提出一种离散猫群优化算法(discrete cat swarm optimization, DCSO),分别设计基于邻域结构的搜寻模式和基于交叉变异的跟踪模式,使算法个体更新能够直接作用于离散调度解。最后,通过大量仿真验证了该算法的有效性。

1 问题描述

车间内 m 台机器需加工 n 个工件,各工件相互独立且交货期已知。加工一个工件 i 需经历多道工序,且每道工序可在多于一台的机器上加工,其加工时间由所分配机器的能力决定。本文要解决的问题是为工序分配合适的机器,并确定机器上工序间的加工顺序,最终使工件最大完工时间和平均提前/拖期时间的加权和最小。

对于该问题,考虑以下假设条件: 1) 工件和机器在初始时刻均处于可用状态; 2) 同一时刻同台机器只可加工一道工序; 3) 工序一旦开始加工则不可被中断; 4) 各工件加工优先级相同; 5) 不考虑机器故障和机器加工前准备时间。

令 TT 表示目标函数, ct_i 表示工件 i 的完工时间, d_i 表示工件 i 的交货期,则 TT 可表示为

$$\min TT = \max_{1 \leq i \leq n} (ct_i) + \frac{\sum_{i=1}^n |ct_i - d_i|}{n}. \quad (1)$$

式(1)中,等号右边第1项表示最大完工时间,第2项表示平均提前/拖期时间。

2 基本猫群优化算法

基本猫群优化算法每次迭代时均先对猫群进行个体划分,然后分别对划分后的两个子群执行搜寻模式和跟踪模式下的操作,对应于全局搜索和局部搜索,从而实现个体更新,直至算法终止。该算法采用连续个体位置编码,各元素在一定范围内取值。搜寻模式下通过“记忆池”机制实现全局搜索,即在算法开始前预先设定搜索记忆池的大小,然后对于当前个体进行个体位置扰动,产生填满记忆池的一系列个体,通过评价记忆池中所有个体位置,选择其中最优位置替代原个体位置,从而完成一个个体的更新。不断循环此过程,直到搜寻模式下所有个体均完成个体更新^[11]。跟踪模式下,算法采用与粒子群算法类似的速度-位置更新公式,并根据全局最优个体来更新每个

个体的位置,如下所示:

$$\mathbf{v}_k^{t+1} = \mathbf{v}_k^t + c \times \text{rand} \times (\mathbf{x}_k^t - \mathbf{x}^*), \quad (2)$$

$$\mathbf{x}_k^{t+1} = \mathbf{x}_k^t + \mathbf{v}_k^{t+1}. \quad (3)$$

其中: t 表示算法当前迭代次数, $\mathbf{x}_k$ 表示第 k 只猫的个体位置, $\mathbf{x}^*$ 表示全局最优个体位置, $\mathbf{v}_k$ 表示第 k 只猫的速度向量, c 表示一个常数, rand 表示 $[0, 1]$ 间的随机数.由于篇幅有限,基本猫群优化算法的步骤可详见文献[11].

3 离散猫群优化算法

3.1 编码方法

本文首先从个体编码上对猫群优化算法进行离散化设计.采用相同长度的前后两段式离散编码,分别对应机器分配方案和工序排序方案.如某车间内包含两台机器和3个工件,每个工件均需完成两道工序,则个体位置向量大小为12,编码方案如图1所示.前段元素为工序所分配机器的编号,按固定顺序进行存储;后段元素为工件编号,其中相同元素值表示同一工件的不同工序,其位置关系表示工序的加工先后.

O_{11}	O_{12}	O_{21}	O_{22}	O_{31}	O_{32}	O_{11}	O_{12}	O_{21}	O_{22}	O_{31}	O_{32}
1	1	2	2	1	2	1	2	2	1	3	3

图1 个体编码

解码时,根据个体后段编码确定工序加工顺序,再根据前段编码找出分配的机器.图1所对应的加工方案为 $O_{11}^1 \rightarrow O_{21}^2 \rightarrow O_{22}^2 \rightarrow O_{12}^1 \rightarrow O_{31}^1 \rightarrow O_{32}^2$,其中 O_{11}^1 表示工件1的第1道工序在机器1上加工,其他工序以此类推.

3.2 种群初始化

对于群智能优化算法,初始种群中解的质量影响算法的收敛速度和求解质量.根据3.1节中个体编码的基本形式,分别对前后两段进行初始化操作.

1) 前段采用文献[18]中的启发式算法,即全局搜索、局部搜索与随机生成3种方式相结合来获得初始机器分配方案.本文算法中个体选择3种方法生成初始机器分配方案的比例为40%、40%和20%.

2) 后段针对每一个已生成的机器分配方案,按照随机生成法获得若干个排序方案,并计算每一个排序方案与机器分配方案的组合所对应的目标值,然后选择最优组合作为一个初始调度解,不断重复该过程,直到生成初始种群为止.

3.3 搜寻模式

基本猫群优化算法搜寻模式中通过预先设定需改变基因的位数来实现个体更新,该参数对算法搜索

具有较大的影响.参数太大,则使算法解的变化范围较大,导致算法变成随机搜索;参数过小,则使算法中解的变化范围较小,削弱算法搜索能力^[13].对此,本文根据柔性作业车间调度问题的特点,采用3种不同邻域结构,并基于此设计新的搜寻模式,以确保算法个体的多样性,扩展算法的搜索空间,提高全局搜索的能力.邻域结构如下:

1) N_1 :在工序排序部分随机选出两个元素值不同的元素,然后互换所选元素的位置.

2) N_2 :在工序排序部分随机选出两个元素,然后将后面元素插入到前面元素之前的位置.

3) N_3 :在机器分配部分随机选出一个元素,该元素对应工序的可选机器应多于一台,然后将此工序分配到其可加工机器集中加工时间最小的机器上.

搜寻模式下搜索方法可描述如下:对于该模式下的每一个个体,首先将记忆池清空,然后随机执行上述一种邻域操作,形成新的个体,并将其填入记忆池;填满后,从记忆池中选出最优个体与原来个体进行比较,若新个体优于原个体,则替换原来个体,否则保持原来个体.不断重复该过程,直到完成该模式下所有个体更新操作.

3.4 跟踪模式

基本猫群优化算法中跟踪模式通过速度-位置公式更新连续个体位置,但本文3.1节中采用离散型个体编码,因而无法直接利用式(2)和(3)进行速度和位置的更新.考虑到车间调度解的离散特性,本文提出一种跟踪模式下适用于柔性作业车间调度问题的离散型猫群个体更新公式,即当前个体通过与全局最优个体间的交叉操作和自身变异操作实现个体更新,如下所示:

$$\bar{\mathbf{x}}_k^{t+1} = \varsigma \otimes f_2(w \otimes f_1(\bar{\mathbf{x}}_k^t), \bar{\mathbf{x}}^*). \quad (4)$$

其中: $\bar{\mathbf{x}}_k$ 表示第 k 只猫对应的离散调度解, $\bar{\mathbf{x}}^*$ 表示当前全局最优解; ς 和 w 分别表示交叉率和变异率,取值范围均为 $[0, 1]$; f_1 表示离散变异操作,在机器分配部分采用单点变异法,在工序排序部分采用逆序变异法,即任选两个位置,然后使所选位置间的元素逆序排序; f_2 表示离散交叉操作,即在机器分配部分采用两点交叉法,在工序排序部分采用文献[19]中的POX交叉法.由式(4)可以看出,个体更新公式可分解为如下两部分:

$$E_k^t = w \otimes f_1(\bar{\mathbf{x}}_k^t) = \begin{cases} f_1(\bar{\mathbf{x}}_k^t), & \text{rand} < w; \\ \bar{\mathbf{x}}_k^t, & \text{rand} \geq w; \end{cases} \quad (5)$$

$$F_k^t = \varsigma \otimes f_2(E_k^t, \bar{\mathbf{x}}^*) = \begin{cases} f_2(E_k^t, \bar{\mathbf{x}}^*), & \text{rand} < \varsigma; \\ E_k^t, & \text{rand} \geq \varsigma. \end{cases} \quad (6)$$

式(5)表示个体按一定概率进行变异操作,代表对自身先前状态的思考;式(6)表示个体按照一定的概率与全局最优解进行交叉操作,以增强算法学习能力。

3.5 自适应模式选择方法

由前述可知,猫群优化算法将种群划分为搜寻模式和追踪模式下的两个子群,分别对应于全局搜索和局部搜索,但基本猫群优化算法中个体划分比例是固定不变的。然而,对于一个好的算法,应该在运行过程中合理协调全局搜索和局部搜索的能力,即运行初期侧重于全局搜索,增强个体多样性,运行后期侧重于局部搜索,增强算法开发能力。本文采用文献[12]中的基于线性混合比率的个体行为模式自适应选择方法,实现对猫群个体的划分,如下所示:

$$MR = MR_{\max} - (MR_{\max} - MR_{\min}) \times t/t_{\max}. \quad (7)$$

其中:MR表示混合比率,即执行搜寻模式的个体占种群个体的数量比;MR_{max}和MR_{min}表示MR的最大值和最小值,分别取为1和0;t和t_{max}分别表示算法当前次数和最大迭代次数。

3.6 局部搜索

为了能够进一步地改善算法结果,基于3.3节中的邻域结构引入一种基于阈值接受法的局部搜索策略,并将其作用于每代当前最优个体。局部搜索策略具体步骤如下。

Step 1: 将当前最优个体作为初始调度解 $\bar{x}$,并令阈值 $\theta > 0$,迭代次数 $q \leftarrow 1$,标记位 $\vartheta \leftarrow 1$,设置最大迭代次数 $q_{\max}$ 。

Step 2: 若 $\vartheta = 1$,则同时执行邻域结构 N_1 和 N_3 ,即 $\bar{x}' \leftarrow N_1(\bar{x}) \cup N_3(\bar{x})$;若 $\vartheta = 0$,则同时执行邻域结构 N_2 和 N_3 ,即 $\bar{x}' \leftarrow N_2(\bar{x}) \cup N_3(\bar{x})$ 。

Step 3: 判断是否满足 $TT(\bar{x}') - TT(\bar{x}) \leq \theta$ 。若是,则 $\bar{x} \leftarrow \bar{x}'$;否则,令 $\vartheta \leftarrow |\vartheta - 1|$ 。

Step 4: 设置 $q \leftarrow q + 1$,并判断是否满足 $q > q_{\max}$ 。若是,则 $\bar{x}' \leftarrow \bar{x}$,转到Step 5;否则,转到Step 2。

Step 5: 局部搜索结束,输出局部最优解 $\bar{x}'$ 。

3.7 算法流程

基于前述,DCSO算法流程如图2所示,并详细阐述如下。

Step 1: 采用离散编码方式,并根据种群初始化方法获得初始种群。

Step 2: 根据自适应模式选择方法计算MR的值,并对猫群进行划分。

Step 3: 判断每个个体所处的模式。若处于搜寻模式,则对其执行搜寻算子;否则,执行跟踪算子。

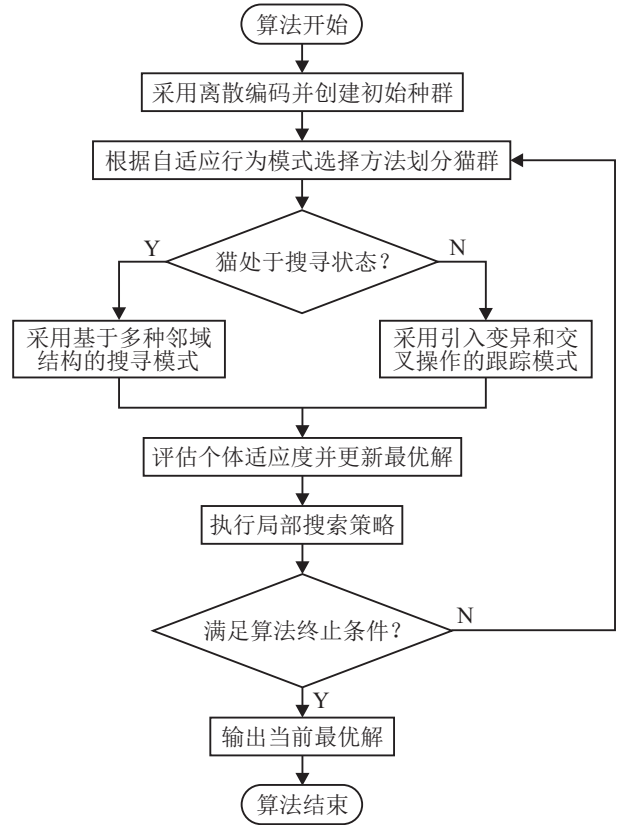


图2 DCSO算法流程

Step 4: 评估个体适应度并更新最优个体。

Step 5: 对当前最优个体执行局部搜索操作。

Step 6: 判断是否满足终止条件。若是,则转到Step 7;否则,转到Step 2。

Step 7: 算法结束并输出结果。

3.8 算法复杂度分析

假设种群规模为 P ,迭代次数为 $t_{\max}$,车间内总工序数为 l ,记忆池大小为SP。从3.7节中的算法流程可以看出,DCSO算法增加的计算量主要发生在个体更新和局部邻域搜索的迭代过程中。

DCSO算法中Step 3的个体更新分为搜寻和跟踪两种模式。每次迭代时,搜寻模式下每个个体根据邻域结构对工序排序或机器分配共需进行SP次调整,因此搜寻模式的最差时间复杂度为 $O(P \times SP \times l)$;跟踪模式下交叉操作最坏需要对工序排序和机器分配各调整一次,变异操作最坏也需对工序排序和机器分配各调整一次,因此跟踪模式的最差时间复杂度为 $O(P \times 4l)$;Step 4中计算个体适应度并更新最优个体的时间复杂度为 $O(P \times l) + O(P)$;Step 5中只针对当前最优个体执行局部搜索操作,每次邻域操作需同时对工序排序和机器分配各调整一次,因此局部搜索的时间复杂度为 $q_{\max} \times O(2l)$ 。

综上所述,DCSO算法迭代 $t_{\max}$ 次的最差时间复杂度为

$$\begin{aligned} & t_{\max} \times [O(P \times SP \times l) + O(P \times 4l) + \\ & O(P \times l) + O(P) + q_{\max} \times O(2l)] = \\ & t_{\max} \times [O(P \times SP \times l) + 5O(P \times l) + \\ & O(P) + q_{\max} \times O(2l)] \approx \\ & t_{\max} \times O(P \times SP \times l). \end{aligned}$$

从上式可以看出,DCSO算法的计算量主要与种群规模、迭代次数、记忆池大小以及总工序数有关。

4 算例分析

算法采用 Fortran 语言编程,在 WinXP 下内存为 2 G 的 VMware Workstation 上运行. 通过多次实验的方法进行参数设置,即种群大小为 300,记忆池大小为 30, w 和 ς 分别取为 0.1 和 0.8,局部搜索最大迭代次数 $q_{\max}$ 为 30. 为了方便比较,采用与文献[1]相同的终止条件,即最大迭代次数为 $10 \times n \times m$. 在 Brandimarte 算例^[20]的基础上进行仿真,各算例中工件交货期 d_i 根据文献[21]中的方法进行设置,表示为

$$d_i = \left(1 + \frac{0.3 \times n}{m}\right) \times \sum_{j=1}^{J_i} p_{ij}. \quad (8)$$

其中: p_{ij} 表示工件 i 的第 j 道工序的加工时间, J_i 表示工件 i 的总工序数.

为了验证本文算法的有效性,将 DCSO 算法与现有文献中的 DHS 算法^[1]、ICSO 算法^[17]、HGWO 算法^[22]、AGA 算法^[23]和 EPABC 算法^[24]进行比较. 对于 AGA 算法、EPABC 算法和 DHS 算法,根据文献[1]中的参数进行设置;对于 HGWO 算法,种群大小为 100,

最大迭代次数为 $10 \times n \times m$,交叉率和变异率分别为 0.8 和 0.2,变邻域搜索和局部搜索的最大迭代次数均为 10;对于 ICSO 算法,种群大小为 300,最大迭代次数为 $10 \times n \times m$,记忆池大小为 30,解龄为 10,变邻域搜索和局部搜索的最大迭代次数均为 10. 表 1 中: avg 表示各算法 10 次运行的平均结果, time 表示各算法 10 次运行的平均时间(单位为 s),粗体表示同一算例下各算法平均结果的最佳值,“—”表示文献中未列出相应数据. 由表 1 中的数据可以看出,DCSO 算法时间明显优于 ICSO 算法,劣于 HGWO 算法,但本文 DCSO 算法能获得最佳结果的个数(6 个)多于其他算法. 图 3 和图 4 分别为本文算法获得的 MK01 和 MK07 算例的甘特图.

此外,本文还验证了 DCSO 算法在求解柔性作业车间工件最大完工时间方面的有效性. 通过多次实验的方法进行参数设置,即种群大小为 200,最大迭代次数为 800,记忆池大小为 30, w 和 ς 分别取为 0.1 和 0.9,局部搜索最大迭代次数 $q_{\max}$ 为 20. 针对 Kacem 算例^[25]和 Brandimarte 算例^[20]的每种情况,算法独立运行 10 次,取最优结果,并与现有文献算法(ICSO^[17]、HGWO^[22]、AL+GA^[25]、LEGA^[26]、Heuristic^[27]、edPSO^[28]、SinGWO^[29]和 NIMASS^[30])的结果进行对比,结果如表 2 和表 3. 其中: BKS 表示各算例的标准解, best 表示 10 次运行的最优结果, time 表示各算法 10 次运行的平均时间(单位为 s),“—”表示文献中未列出相应数据.

表 1 DCSO 算法与其他算法的对比结果

算例	$n \times m$	AGA		EPABC		DHS		HGWO		ICSO		DCSO	
		avg	time	avg	time	avg	time	avg	time	avg	time	avg	time
MK1	10 × 6	46.00	—	46.20	—	43.50	—	45.50	33.2	45.70	126.3	43.20	80.0
MK2	10 × 6	33.00	—	32.00	—	31.00	—	32.05	35.0	32.10	125.0	29.00	87.6
MK3	15 × 8	225.00	—	225.00	—	225.00	—	211.45	300.6	213.80	1 078.7	210.70	482.8
MK4	15 × 8	77.00	—	79.00	—	77.00	—	79.20	142.5	84.80	494.8	75.90	276.1
MK5	15 × 4	222.00	—	223.00	—	218.00	—	228.40	89.6	222.20	311.4	214.80	165.4
MK6	10 × 15	70.00	—	71.00	—	69.00	—	80.00	378.1	80.10	1 326.2	78.40	569.3
MK7	20 × 5	167.00	—	165.00	—	164.00	—	161.60	147.9	166.70	522.8	160.50	286.5
MK8	20 × 10	701.00	—	700.00	—	690.00	—	730.80	1 010.1	726.60	3 746.0	716.90	1 310.5
MK9	20 × 10	398.00	—	397.00	—	396.00	—	437.30	1 133.9	434.90	4 154.7	417.20	1 413.8
MK10	20 × 15	257.20	—	263.00	—	246.00	—	289.73	1 748.9	303.6	6 313.5	272.80	2 322.7

表 2 Kacem 算例的计算结果

算例	$n \times m$	BKS	ICSO		HGWO		AL+GA		LEGA		Heruistic		edPSO		DCSO	
			best	time	best	time	best	time	best	time	best	time	best	time	best	time
Kacem1	4 × 5	11	11	6.8	11	6.0	—	—	11	1.2	11	0.0	11	0.4	11	12.4
Kacem2	8 × 8	14	14	18.6	14	15.3	15	—	—	—	15	0.1	17	5.7	14	27.2
Kacem3	10 × 7	11	11	20.6	11	17.1	—	—	11	2.6	13	0.1	—	—	11	30.1
Kacem4	10 × 10	7	7	22.2	7	18.1	7	—	7	2.8	7	0.2	8	7.3	7	32.2
Kacem5	15 × 10	11	14	46.0	13	41.9	23	—	12	4.9	12	0.5	—	—	12	65.4

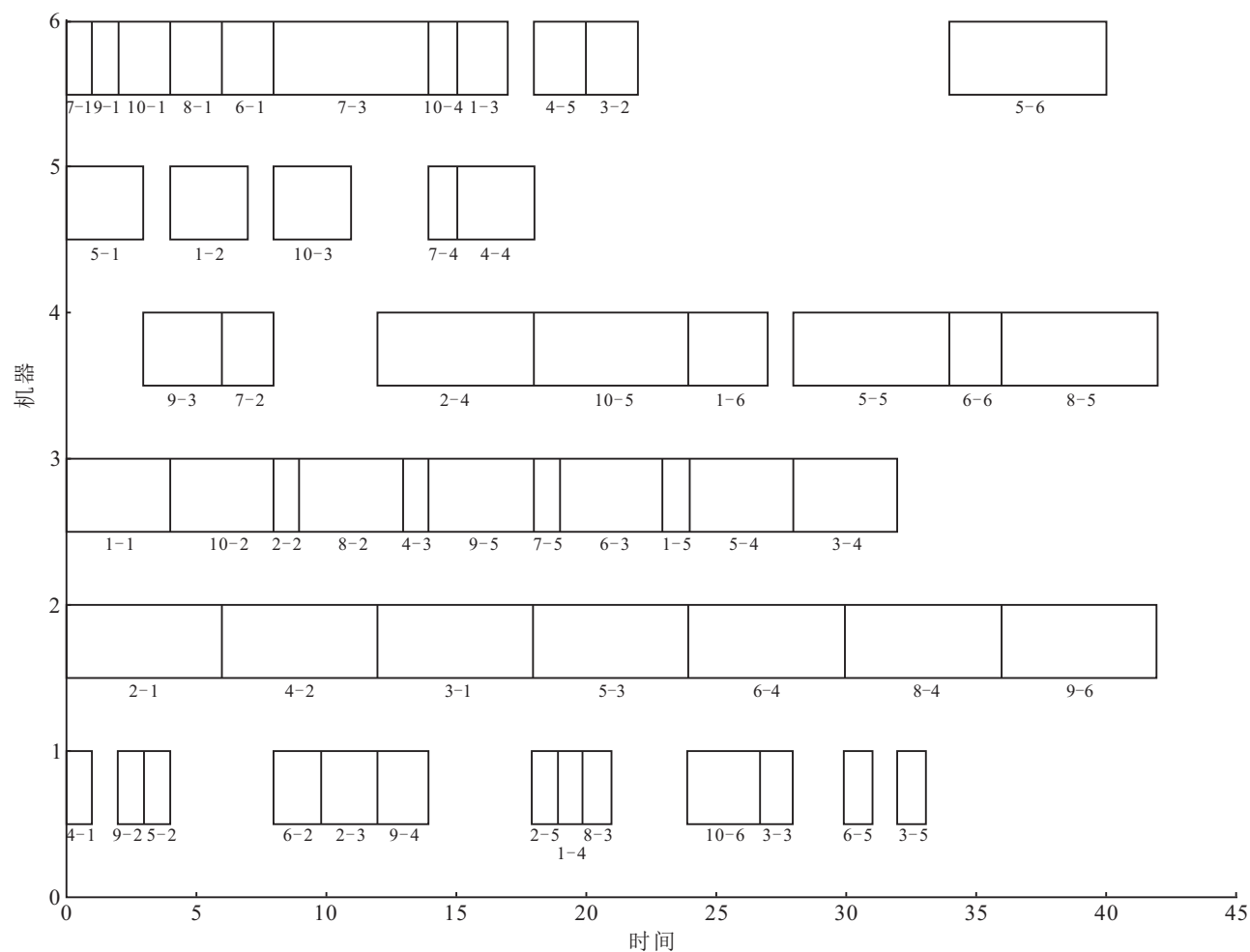


图3 算例MK01的甘特图

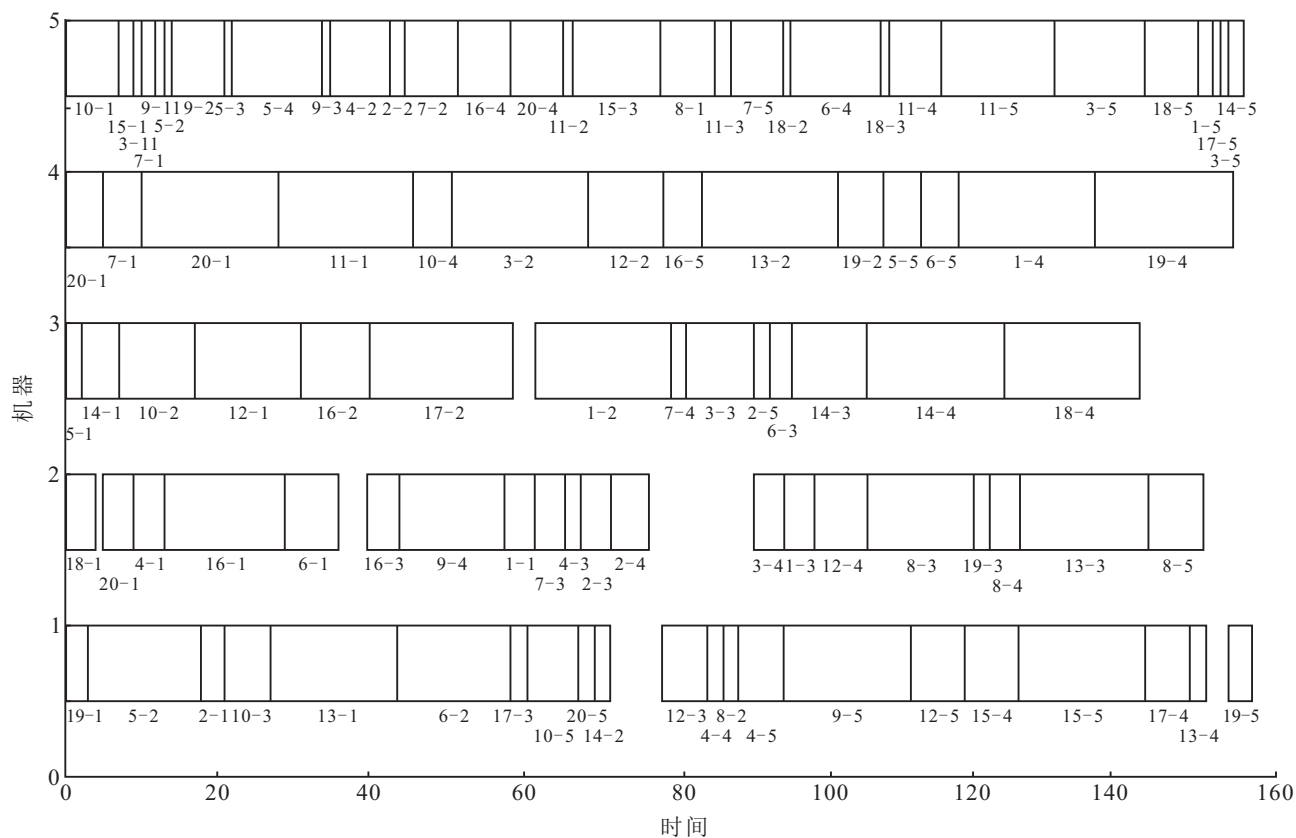


图4 算例MK07的甘特图

表3 Brandimarte算例的计算结果

算例	$n \times m$	BKS	ICSO		HGWO		LEGA		Heuristic		edPSO		SinGWO		NIMASS		DCSO	
			best	time	best	time	best	time	best	time	best	time	best	time	best	time	best	time
MK1	10 × 6	36	40	46.4	40	37.4	40	3.3	42	0.1	41	—	42	5.2	40	1.1	40	53.2
MK2	10 × 6	24	28	49.6	29	40.6	29	3.6	28	0.2	26	—	31	5.7	28	1.8	27	55.3
MK3	15 × 8	204	204	219.0	204	171.4	—	—	204	0.5	207	—	204	21.4	204	4.0	204	167.5
MK4	15 × 8	48	67	98.1	65	78.8	67	6.1	75	0.2	65	—	68	10.4	65	2.4	63	97.7
MK5	15 × 4	168	176	124.5	175	100.1	176	7.2	179	0.2	171	—	179	13.3	177	2.2	173	111.8
MK6	10 × 15	33	73	219.6	79	170.6	67	10.7	69	0.5	61	—	68	20.9	67	4.5	67	165.5
MK7	20 × 5	133	149	120.3	149	94.8	147	6.9	149	0.4	173	—	155	12.2	144	3.7	144	117.6
MK8	20 × 10	523	523	450.3	523	345.3	523	19.1	555	0.7	523	—	523	41.4	523	7.7	523	283.8
MK9	20 × 10	299	334	498.3	325	381.3	320	20.4	342	0.9	307	—	340	46.4	312	11.4	311	301.1
MK10	20 × 15	165	259	509.6	253	398.2	229	27.7	242	1.2	312	—	280	45.7	229	15.3	235	318.1

为了充分体现本文DCSO算法的有效性,表4对各算法的平均百分比偏差(average relative percentage deviation, ARPD)进行比较,其计算公式如下所示:

$$APRD = \sum_{l=1}^{NI} \frac{100 \times (\text{Best} - \text{BKS})}{\text{BKS}} / NI. \quad (9)$$

其中:NI表示算例的个数,Best表示各算法所获得的最佳结果.综合表2~表4的数据可以看出,本文DCSO算法运行时间只优于ICSO算法和HGWO算法,但其所获得的计算结果优于其他算法.

表4 ARPD值的对比结果

算法	算例个数	现有算法ARPD	本文算法ARPD	改善百分比
ICSO	15	20.1	15.0	5.1
HGWO	15	20.2	15.0	5.2
AL+GA	3	38.7	3.0	35.7
LEGA	13	18.8	17.3	1.5
Heuristic	15	21.3	15.0	6.3
edPSO	13	23.3	16.6	6.7
SinGWO	10	30.0	21.6	8.4
NIMASS	10	22.3	21.6	0.7

综上所述,本文所提出的DCSO算法在求解FJSP问题方面具有有效性.

5 结 论

本文以工件最大完工时间和平均提前/拖期时间为优化目标求解带交货期的柔性作业车间调度问题,提出了一种离散型猫群优化算法.

1) 针对问题的特点,设计了两段式离散编码和基于启发式算法的种群初始化方法;采用基于线性混合比率的行为模式自适应选择方法,以协调算法的全局搜索能力和局部搜索能力;采用基于多样化搜寻算子的搜寻模式,并设计了3种邻域结构作为该模式下的搜寻方法;提出了一种跟踪模式下的离散型个体更新公式,使算法能够直接在离散调度空间内运行.此外,在算法中引入了局部搜索策略,对算法性能进行了进一步的改善.

2) 基于柔性作业车间调度问题的基准算例,测试

了本文算法的性能.仿真结果显示,DCSO算法在求解FJSP问题方面具有有效性.

3) 未来将把CSO算法与其他元启发式算法相结合,设计出一种更高效的算法,并将其应用于更复杂的车间调度问题.

参考文献(References)

[1] Gao K Z, Suganthan P N, Pan Q K, et al. Discrete harmony search algorithm for flexible job shop scheduling problem with multiple objectives[J]. Journal of Intelligent Manufacturing, 2016, 27(2): 363-374.

[2] Mousakhani M. Sequence-dependent setup time flexible job shop scheduling problem to minimise total tardiness[J]. International Journal of Production Research, 2013, 51(12): 3476-3487.

[3] Scrich C R, Armentano V A, Laguna M. Tardiness minimization in a flexible job shop: A tabu search approach[J]. Journal of Intelligent Manufacturing, 2004, 15(1): 103-115.

[4] Wu Z, Weng M X. Multiagent scheduling method with earliness and tardiness objectives in flexible job shops[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2005, 35(2): 293-301.

[5] Türkylmaz A, Bulkan S. A hybrid algorithm for total tardiness minimisation in flexible job shop: Genetic algorithm with parallel VNS execution[J]. International Journal of Production Research, 2015, 53(6): 1832-1848.

[6] Na H, Park J. Multi-level job scheduling in a flexible job shop environment[J]. International Journal of Production Research, 2014, 52(13): 3877-3887.

[7] Huang R H, Yang C L, Cheng W C. Flexible job shop scheduling with due window — A two-pheromone ant colony approach[J]. International Journal of Production Economics, 2013, 141(2): 685-697.

[8] 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.

(Liu X X, Xie L Y, Tao Z, et al. Research on multi-objective scheduling optimization for flexible job shop[J]. Journal of Northeastern University: Natural Science, 2008, 29(3): 362-365.)

- [9] 王云, 谭建荣, 冯毅雄, 等. 基于SPEA的多目标柔性作业车间调度方法[J]. 中国机械工程, 2010, 21(10): 1167-1172.
(Wang Y, Tan J R, Feng Y X, et al. Multi-objective flexible job-shop scheduling based on strength pareto evolutionary algorithm[J]. China Mechanical Engineering, 2010, 21(10): 1167-1172.)
- [10] Vilcot G, Billaut J C. A tabu search algorithm for solving a multicriteria flexible job shop scheduling problem[J]. International Journal of Production Research, 2011, 49(23): 6963-6980.
- [11] Chu S C, Tsai P W. Computational intelligence based on the behavior of cats[J]. International Journal of Innovative Computing, Information and Control, 2007, 3(1): 163-173.
- [12] Pradhan P M, Panda G. Solving multiobjective problems using cat swarm optimization[J]. Expert Systems with Applications, 2012, 39(3): 2956-2964.
- [13] 王光彪, 杨淑莹, 冯帆, 等. 基于猫群算法的图像分类研究[J]. 天津理工大学学报, 2011, 27(5): 35-39.
(Wang G B, Yang S Y, Feng F, et al. Research of image classification based on cat swarm optimization[J]. Journal of Tianjin University of Technology, 2011, 27(5): 35-39.)
- [14] Lin K C, Huang Y H, Hung J C, et al. Feature selection and parameter optimization of support vector machines based on modified cat swarm optimization[J]. International Journal of Distributed Sensor Networks, 2015, 11(7): 1-9.
- [15] Skoullis V I, Tassopoulos I X, Beligiannis G N. Solving the high school timetabling problem using a hybrid cat swarm optimization based algorithm[J]. Applied Soft Computing, 2017, 52: 277-289.
- [16] 刘琼, 范正伟, 张超勇, 等. 基于多目标猫群算法的混流装配线排序问题[J]. 计算机集成制造系统, 2014, 20(2): 333-342.
(Liu Q, Fang Z W, Zhang C Y, et al. Mixed model assembly line sequencing problem based on multi-objective cat swarm optimization[J]. Computer Integrated Manufacturing Systems, 2014, 20(2): 333-342.)
- [17] 姜天华. 猫群优化算法求解柔性作业车间调度问题[J]. 计算机工程与应用, 2018, 54(23): 259-263.
(Jiang T H. Cat swarm optimization for solving the flexible job shop scheduling problem[J]. Computer Engineering and Applications, 2018, 54(23): 259-263.)
- [18] Zhang G, Gao L, Shi Y. An effective genetic algorithm for the flexible job-shop scheduling problem[J]. Expert Systems with Applications, 2011, 38(4): 3563-3573.
- [19] Demir Y, Isleyen S K. An effective genetic algorithm for flexible job-shop scheduling with overlapping in operations[J]. International Journal of Production Research, 2014, 52(13): 3905-3921.
- [20] Brandimarte P. Routing and scheduling in a flexible job shop by tabu search[J]. Annals of Operations Research, 1993, 41(3): 157-183.
- [21] Demirkol E, Mehta S, Uzsoy R. Benchmarks for shop scheduling problems[J]. European Journal of Operational Research, 1998, 109(1): 137-141.
- [22] 姜天华. 混合灰狼优化算法求解柔性作业车间调度问题[J]. 控制与决策, 2018, 33(3): 503-508.
(Jiang T H. Flexible job shop scheduling problem with hybrid grey wolf optimization algorithm[J]. Control and Decision, 2018, 33(3): 503-508.)
- [23] Gholami M, Zandieh M. Integrating simulation and genetic algorithm to schedule a dynamic flexible job shop[J]. Journal of Intelligent Manufacturing, 2009, 20(4): 481-498.
- [24] Wang L, Zhou G, Xu Y, et al. An enhanced Pareto-based artificial bee colony algorithm for the multi-objective flexible job-shop scheduling[J]. The International Journal of Advanced Manufacturing Technology, 2012, 60(6): 1111-1123.
- [25] Kacem I, Hammadi S, Borne P. Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2002, 32(1): 1-13.
- [26] Ho N B, Tay J C, Lai E M K. An effective architecture for learning and evolving flexible job-shop schedules[J]. European Journal of Operational Research, 2007, 179(2): 316-333.
- [27] Ziaee M. A heuristic algorithm for solving flexible job shop scheduling problem[J]. The International Journal of Advanced Manufacturing Technology, 2014, 71(1/2/3/4): 519-528.
- [28] Nouiri M, Bekrar A, Jemai A, et al. An effective and distributed particle swarm optimization algorithm for flexible job-shop scheduling problem[J]. Journal of Intelligent Manufacturing, 2018, 29(3): 603-615.
- [29] 姜天华. 基于灰狼优化算法的低碳车间调度问题[J]. 计算机集成制造系统, 2018, 24(10): 2428-2435.
(Jiang T H. Low-carbon workshop scheduling problem based on grey wolf optimization, Computer Integrated Manufacturing Systems, 2018, 24(10): 2428-2435.)
- [30] Xiong W, Fu D. A new immune multi-agent system for the flexible job shop scheduling problem[J]. Journal of Intelligent Manufacturing, 2018, 29(4): 857-873.

作者简介

姜天华(1983—), 男, 讲师, 博士, 从事生产调度和智能算法的研究, E-mail: jth1127@163.com;

邓冠龙(1985—), 男, 副教授, 博士, 从事生产调度和智能算法等研究, E-mail: dalag@163.com;

朱惠琦(1986—), 女, 讲师, 博士, 从事物流管理与调度的研究, E-mail: 22849234@qq.com.

(责任编辑: 齐 霖)