

改进教学优化算法求解执行器配置与生产调度协同优化问题

谭伟华^{1,2}, 吴亮红¹, 李哲², 袁小芳^{2†}

(1. 湖南科技大学 信息与电气工程学院, 湖南 湘潭 411201; 2. 湖南大学 电气与信息工程学院, 湖南 长沙 410082)

摘要: 在实际柔性作业车间中, 执行器配置与生产调度的高效协同有利于提高生产决策的全局性, 从而提升车间的柔性加工能力. 针对执行器配置与生产调度的多目标协同优化问题, 以最小化综合生产成本和完工时间为优化目标, 构建了混合整数规划模型, 使得小规模问题通过可以Gurobi精确求解. 本文提出了一种两阶段离散教学优化算法, 设计了学习强度自适应调整方法和改进关键工序移动策略, 以提升算法的效率和多目标平衡搜索能力. 通过仿真实验, 分析验证了所提协同优化方法的优越性和所提算法的有效性.

关键词: 柔性作业车间调度; 资源配置; 多目标优化; 教学优化算法

引用格式: 谭伟华, 吴亮红, 李哲, 等. 改进教学优化算法求解执行器配置与生产调度协同优化问题. 控制理论与应用, 2026, 43(4): 865 – 873

DOI: 10.7641/CTA.2024.40224

Improved teaching-learning-based optimization algorithm for collaborative optimization of end-effector allocation and production scheduling

TAN Wei-hua^{1,2}, WU Liang-hong¹, LI Zhe², YUAN Xiao-fang^{2†}

(1. College of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan Hunan 411201, China;
2. College of Electrical and Information Engineering, Hunan University, Changsha Hunan 410082, China)

Abstract: In the practical flexible job-shop, the efficient collaboration between end-effector allocation and production scheduling benefits providing global decision-making, thus facilitating the flexibility of the workshop. To address the collaborative optimization problem of end-effector allocation and production scheduling, a mixed-integer programming model is constructed to minimize the comprehensive production cost and makespan, enabling the accurate solution of small-scale problems via Gurobi. In addition, a dual-stage discrete teaching-learning-based optimization algorithm is proposed, incorporating a distance-based adaptive adjustment strategy for learning coefficient and an improved neighborhood search based on critical operation movement to facilitate efficiency and objective balancing. Through simulation experiments, the superiority of the proposed collaborative optimization method and the effectiveness of the proposed algorithm are analyzed and validated.

Key words: flexible job-shop scheduling; resource allocation; multi-objective optimization; teaching-learning-based optimization algorithm

Citation: TAN Weihua, WU Lianghong, LI Zhe, et al. Improved teaching-learning-based optimization algorithm for collaborative optimization of end-effector allocation and production scheduling. *Control Theory & Applications*, 2026, 43(4): 865 – 873

1 引言

在实际柔性作业车间中, 加工设备常常配置多个不同功能的执行器^[1-2], 以适应多种加工需求. 随着装备技术的进步, 执行器和设备本体普遍采用模块化设

计^[3]. 通过标准化的软硬件接口, 可以实现执行器与设备本体的灵活适配, 从而快速、低成本地重构加工设备功能, 有效提升加工设备对多种产品类型的适应性. 可见, 执行器的合理配置对提升柔性作业车间的

收稿日期: 2024-04-18; 录用日期: 2024-11-22.

†通信作者. E-mail: yuanxiaofang@hnu.edu.cn; Tel.: +86 13873195923.

本文责任编辑: 王凌.

国家重点研发计划项目(2021YFB3301800), 国家自然科学基金项目(62373146), 湖南省自然科学基金项目(2022JJ30265), 湖南省科技人才托举工程项目(2022TJ-Q03)资助.

Supported by the National Key Research and Development Program of China (2021YFB3301800), the National Natural Science Foundation of China (62373146), the Natural Science Foundation of Hunan Province (2022JJ30265) and the Young Talent of Lifting Engineering for Science and Technology in Hunan Province (2022TJ-Q03).

加工能力起到关键作用。

为保障生产决策的高效可行,需考虑执行器的属性特征。针对执行器数量有限的情况,周光辉等^[4]研究了执行器分配和流转方法,降低了加工成本和碳排放量。Li等^[5]考虑了执行器机械性能与加工工艺参数的关联特性,提出了一种节能集成调度方法。An等^[6]将执行器退化引起的维护作业纳入生产过程中,研究了预测性维护与生产调度协同优化问题。Zhang等^[7]研究了考虑执行器切换时间的生产调度问题。然而,现有研究对执行器配置与生产调度的耦合特性考虑不足,难以精准描述生产指标随配置方案的变化情况,且主要聚焦于并行机^[8-10]、无关并行机^[7,11]等生产环境,在柔性作业车间中的研究尚有待深入。

教学优化算法^[12]具有结构简单、参数少、性能优异等优点^[13],自提出以来被广泛用于求解各类复杂的车间调度问题^[14-18]。目前,教学优化算法求解生产调度问题主要通过定义离散空间到连续空间的映射关系,来实现解的编码和后续进化操作^[19-21]。然而,由于离散空间的结构信息难以反映在连续空间中,基于映射关系的算法结构在收敛速度和解的质量方面的明显不足。同时,如何利用问题知识引导教学优化算法进行高效搜索,亦是本领域的热点问题。

本文面向柔性作业车间生产需求,研究了执行器配置与生产调度协同优化方法,通过协同决策执行器配置、工序分配和加工顺序,旨在统筹优化综合生产成本和完工时间;并提出一种两阶段离散教学优化算法,设计了基于距离自适应调整机制的教学算子和基于改进关键工序移动的邻域搜索,有效提升了算法的综合性能,为模型的高效求解提供有力支撑。

2 问题描述及数学模型构建

2.1 问题描述

本文研究的执行器配置与生产调度协同优化 (collaborative optimization of end-effector allocation and production scheduling, COEAPS) 问题如图 1 所示,具体可以描述为:有待加工工件集 J , 工件 $i (i \in J)$ 包含工序集 O_i 且同工件的各工序遵循预设的工艺路线; 车间有设备本体集 M 和执行器集 K , 执行器 $k \in K$ 可分配到任意设备本体 $m \in M$, 设备本体 m 的插槽容量为 Q_m , 且工序 $O_{ij} (j \in O_i)$ 仅可在其可选执行器集 $K_{ij} (K_{ij} \subseteq K)$ 中选择一个进行加工。该问题还满足以下假设: 所有的设备和原材料在初始时刻即可使用; 加工过程不可中断; 运输时间忽略不计, 加工区有足够的缓冲区。

在 COEAPS 中, 执行器配置将影响设备功能和加工性能指标, 进而约束可行的工序分配和加工顺序。因此, COEAPS 包含 3 个耦合的决策问题: 执行器配置到设备本体、工序分配到执行器、工序加工顺序。为了兼顾生产成本和生产效率指标, COEAPS 的优化目标是最小化综合生产成本和完工时间。该数学模型的符

号及定义如表 1 所示。

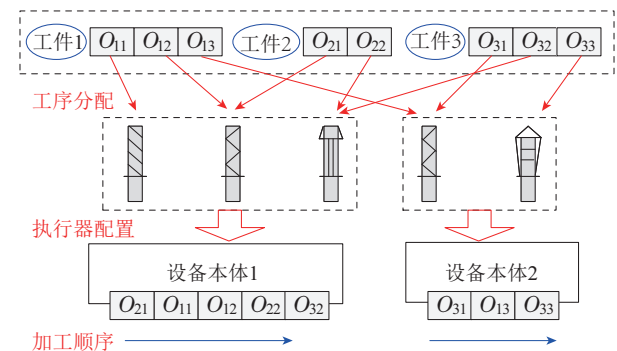


图 1 COEAPS问题示意图

Fig. 1 Diagram of COEAPS problem

表 1 变量符号声明

Table 1 Notations of COEAPS

符号	定义
J	待加工工件集
O_i	工件 i 的工序集
M	设备本体集
K	执行器集
i, i'	工件编号, $i \in J$
j, j'	工序编号, $j \in O_i$
m	设备本体编号, $m \in M$
k	执行器编号, $k \in K$
δ_m	设备本体 m 上的工序集
P_m	设备本体 m 的待机功率
Q_m	设备本体 m 的插槽容量
O_{ij}	工件 i 的第 j 个工序
c_{ij}	O_{ij} 的完成时间
K_{ij}	O_{ij} 的适配执行器集, $K_{ij} \subseteq K$
$T_{ij,k,m}$	执行器 k 在设备本体 m 加工 O_{ij} 的加工时间
$E_{ij,k,m}$	执行器 k 在设备本体 m 加工 O_{ij} 的加工能耗
$v_{ij,k}$	执行器 k 加工 O_{ij} 损耗量
$\xi_e/\xi_t/\xi_m$	能耗、执行器损耗和设备本体使用的成本转换系数
L	足够大的正实数
$X_{ij,k,m}$	若 O_{ij} 由执行器 k 在设备本体 m 上加工, 则 $X_{ij,k,m} = 1$; 否则 $X_{ij,k,m} = 0$
$Y_{ij,m,p}$	若 O_{ij} 在设备本体 m 的第 p 个位置加工, 则 $Y_{ij,m,p} = 1$; 否则 $Y_{ij,m,p} = 0$
$Z_{k,m}$	若执行器 k 配置在设备本体 m 上, 则 $Z_{k,m} = 1$; 否则 $Z_{k,m} = 0$
CPC	综合生产成本
C_{\max}	完工时间

2.2 数学模型

COEAPS 的目标函数表示如式(1)所示:

$$\text{minimize} \begin{cases} f_1 = \text{CPC}, \\ f_2 = C_{\max}, \end{cases} \quad (1)$$

约束条件如下:

$$\sum_{k \in K} \sum_{m \in M} X_{ij,k,m} = 1, \quad i \in J, j \in O_i, \quad (2)$$

$$X_{ij,k,m} \neq X_{i'j',k,m'}, i, i' \in J; j, j' \in O_i; \\ k \in K; m, m' \in M, \quad (3)$$

$$X_{ij,k,m} = 0, \\ i \in J, j \in O_i, k \notin K_{ij}, m \in M, \quad (4)$$

$$\sum_{m \in M} \sum_{p \in \delta_m} Y_{ij,m,p} = 1, i \in J, j \in O_i, \quad (5)$$

$$\sum_{i \in J} \sum_{j \in O_i} Y_{ij,m,p} \leq 1, m \in M, p \in \delta_m, \quad (6)$$

$$\sum_{i \in J} \sum_{j \in O_i} Y_{ij,m,p} \geq \sum_{i' \in J} \sum_{j' \in O_{i'}} Y_{i'j',m,p+1}, \\ m \in M; p, p+1 \in \delta_m, \quad (7)$$

$$\sum_{k \in K} Z_{k,m} \leq Q_m, m \in M, \quad (8)$$

$$\sum_{k \in K} X_{ij,k,m} = \sum_{p \in \delta_m} Y_{ij,m,p}, \\ i \in J, j \in O_i, m \in M, \quad (9)$$

$$Z_{k,m} \geq X_{ij,k,m}, \\ i \in J, j \in O_i, k \in K, m \in M, \quad (10)$$

$$c_{ij} \geq \sum_{k \in K} (T_{ij,k,m} X_{ij,k,m}) + c_{i(j-1)}, \\ i \in J; (j-1), j \in O_i; m \in M, \quad (11)$$

$$c_{ij} \geq \sum_{k \in K} (T_{ij,k,m} X_{ij,k,m}), \\ i \in J, j \in O_i, m \in M, \quad (12)$$

$$c_{i'j'} + L[1 - \sum_{k \in K} (Y_{ij,m,p} Y_{i'j',m,p+1})] - \\ \sum_{k \in K} (T_{i'j',k,m} X_{i'j',k,m}) \geq c_{ij}, \\ i, i' \in J; j \in O_i; j' \in O_{i'}; m \in M; \\ p, p+1 \in \delta_m, \quad (13)$$

其中:约束(2)表示工序只被加工一次;约束(3)表示执行器仅配置到一个设备本体;约束(4)表示工序只能在其可选执行器上完成加工;约束(5)表示工序只能占据一个加工位置;约束(6)表示预设的加工位置最多只能被一个工序占用;约束(7)表示加工位置最多被选择一次且不存在中间位置空缺的情况;约束(8)表示配置在设备本体的执行器不能超过其插槽容量;约束(9)–(10)为决策变量间的一致性约束,避免出现矛盾决策;约束(11)为工件的工艺路线约束;约束(12)表示工序的加工开始时间不早于0时刻;约束(13)表示设备不能同时加工多个工序。

2.3 综合生产成本目标函数

根据实际生产情况,本文在综合生产成本考虑了3个方面的因素:能耗成本、执行器损耗成本和加工设备折旧成本,可由式(14)计算得到。下面对各部分成本的量化过程进行描述:

$$CPC = PC_e + PC_t + PC_m. \quad (14)$$

a) 能耗成本.

设备本体是生产车间的主要耗能单元,本文在能耗成本的核算中考虑了设备本体总能耗带来的成本。

实际生产中,设备本体的总能耗主要包括加工能耗和待机能耗^[22].其中,加工能耗由行器、工序及设备本体共同决定,用 $E_{ij,k,m}$ 表示执行器 k 在设备本体 m 加工 O_{ij} 的加工能耗;待机能耗可以由待机功率 P_m 与待机时长相乘得到.因此,能耗成本由能耗量乘以能耗成本转换系数求得,如式(15)所示:

$$PC_e = \xi_e \left\{ \sum_{i \in J} \sum_{j \in O_i} \sum_{k \in K} \sum_{m \in M} (E_{ij,k,m} X_{ij,k,m}) + \sum_{m \in M} P_m [C_{\max} - \sum_{i \in J} \sum_{j \in O_i} \sum_{k \in K} (E_{ij,k,m} X_{ij,k,m})] \right\}. \quad (15)$$

b) 执行器损耗成本.

使用执行器加工将带来一定的执行器损耗,由工序的工艺属性和执行器自身属性确定,用 $v_{ij,k}$ 表示执行器 k 加工工序 O_{ij} 的损耗量,在实际应用中可通过模型或数据驱动方法获得 $v_{ij,k}$ 值^[23–24].因此,执行器损耗成本可由式(16)计算得到,即

$$PC_t = \sum_{k \in K} \xi_t \left[\sum_{i \in J} \sum_{j \in O_i} (v_{ij,k} \sum_{m \in M} X_{ij,k,m}) \right]. \quad (16)$$

c) 加工设备折旧成本.

加工设备运行会使得其零部件产生磨损、老化,将带来一定的折旧成本.在实际应用中,加工设备的折旧成本通常不可忽略.本文认为加工设备折旧成本与完工时间成正比,由式(17)算法得到,即

$$PC_m = \sum_{m \in M} \xi_m C_{\max}. \quad (17)$$

2.4 完工时间目标函数

完工时间定义为最后一个工序加工完成的时间.根据定义,完工时间 C_{\max} 可由式(18)计算得到,即

$$C_{\max} = \max\{C_{ij}\}. \quad (18)$$

3 两阶段离散教学优化算法

为了对大规模的COEAPS问题进行高效求解,需要确定合适的离散编码形式,并提升算法的双目标平衡搜索能力.本文提出了一种两阶段离散教学优化算法(dual-stage discrete teaching-learning-based optimization, DDTLO),其整体流程如算法1^[25](见表2)所示.

3.1 编码与解码策略

编码方式需保证解的可行性并便于算法的进化.针对COEAPS问题,本文设计了一种三行向量的编码方式.三行向量包括:工序顺序向量OpSeq、工序分配向量OpAssign和执行器配置向量EffectorAllocate.图2所示为一个3工件(共8工序)、4执行器、2设备本体的编码,这些向量的具体描述如下:

a) OpSeq向量:该向量用于表示工序的加工顺序,其长度等于待加工工序的总数,为保证同工件的工序顺序约束^[26],元素值为工件索引,用于表示该工件的

工序,同工件的所有工序均用该工件索引表示,并依据工件索引从左至右的出现顺序区分同工件的不同工序,如:图2的OpSeq向量中第6个元素表示工件3的第2个工序;

b) OpAssign向量: 该向量用于表示为工序分配的执行器,其长度等于待加工工序的总数,向量各元素

与工序一一对应,元素值为执行器索引,用于表示为对应工序分配的执行器;

c) EffectorAllocate向量: 该向量用于表示执行器分配到设备本体的情况,其长度等于执行器的总数,向量元素与执行器一一对应,元素值为设备本体索引,用于表示对应执行器将配置到的设备本体。

表 2 算法 1: 离散教学优化算法的整体流程伪代码

Table 2 Algorithm 1: Pseudocode of the proposed discrete teaching-learning-based optimization algorithm

```

输入: 种群数量  $n_{pop}$ , 算法终止条件
输出: Pareto 前沿解集  $F$ 
1 随机初始化得到种群  $P$ 
2 while 未达到终止条件 do
3    $C \leftarrow \text{DistAdpteTL}(P)$       % 基于距离自适应调整机制的教学算子, 参见第3.2小节
4    $P \leftarrow \text{MergeDiscardRep}(P, C)$   % 合并种群并去掉重复解
5    $P \leftarrow \text{NeighborSearch}(P)$     % 基于改进关键工序移动的邻域搜索, 参见第3.3小节
6    $Ps \leftarrow \text{NonDominateSort}(P)$   % 将  $P$  进行非支配排序[25], 确定 Pareto 前沿存入  $Ps$ 
7    $P = Ps(1:n_{pop})$ 
8 end while
9  $Ps \leftarrow \text{NonDominateSort}(P)$ 
10  $F \leftarrow Ps\{1\}$       % 将  $Ps$  中的 Pareto 前沿存入  $F$ 
    
```

本文采用“主动解码”^[27]的方式进行解码,其过程可以描述为:按照OpAssign向量和EffectorAllocate向量确定工序-执行器-设备本体的对应关系,从而确定各工序相关的加工参数;按照OpSeq向量从左至右依次安排工序,在不影响其他工序的开始时间和不违反约束条件的前提下将工序安排在最早的开始时间。

3.2 基于距离自适应调整机制的教学算子

教学算子是教学优化算法的核心,包括“教授”和“学习”两个阶段.种群中的解根据目标函数值被分为教师解和学生解,学生解通过从教师解和其他学生解中获得部分编码片段的方式(即“学习过程”(实现更新. DDTLO算法采用教学算子对OpAssign向量和Ef-

factorAllocate 向量施加扰动.采用多点交叉 (multi-point crossover, MPX) 操作实现编码片段的嵌入,MPX操作示意图如图3所示.

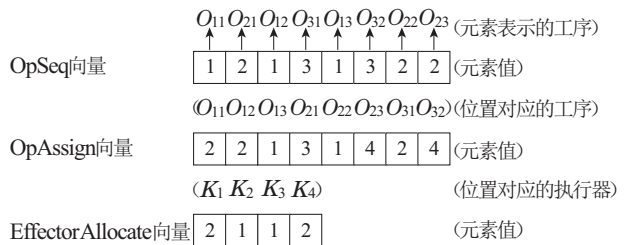


图 2 编码示例

Fig. 2 Illustration of coding method

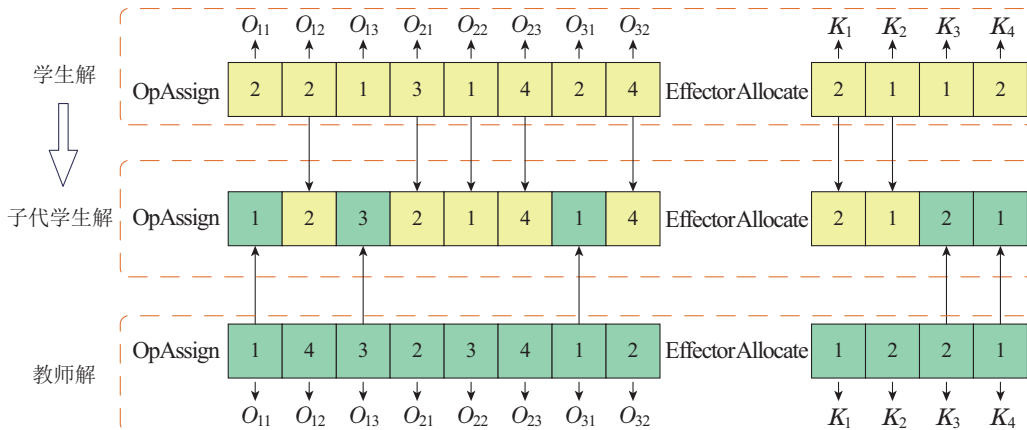


图 3 MPX操作示意图

Fig. 3 Diagram of MPX operator

学习强度是教学优化算法的重要参数, 表示学生解从教师解中获得元素的比例. 传统的教学优化算法缺乏学习强度的有效调控机制, 导致种群多样性难以保持. 为此, 在DDTLO算法的教学算子中设计了距离自适应调整机制, 即通过将目标空间的归一化距离线性映射到学习强度的取值范围上, 根据学生解与所选教师解的实际距离计算学习强度, 原理如图4所示, 图中 d , d_{\max} 和 d_{\min} 分别为学生解到其所选教师解、所有教师解的最大和最小归一化距离. 学习强度 ρ 通过式(19)计算, 式中 $[\rho_{\min}, \rho_{\max}]$ 为 ρ 的取值范围. 通过 ρ 与元素总数相乘并向上取整, 可得到MPX操作的元素数量, 即

$$\rho = \frac{(d - d_{\min}) \times (\rho_{\max} - \rho_{\min})}{d_{\max} - d_{\min}}. \quad (19)$$

基于距离自适应调整机制的教学算子整体流程图如算法2(见表3)所示.

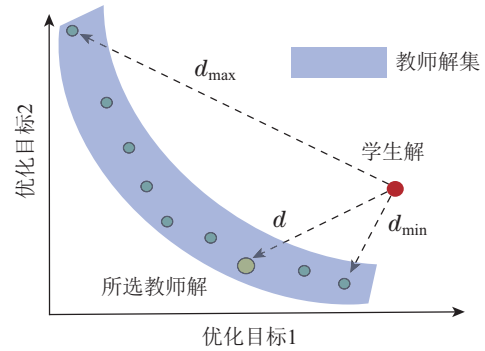


图4 距离自适应调整机制原理

Fig. 4 Diagram of distance-based adaptive adjustment strategy

表3 算法2: 基于距离自适应调整机制教学算子伪代码

Table 3 Algorithm 2: Pseudocode of the teaching operator based on distance-adaptive adjustment mechanism

```

输入: 种群 $P$ , 学习强度取值范围 $[\rho_{\min}, \rho_{\max}]$ 
输出: 子代种群 $C$ 
1  $[\mathbf{Te}, \mathbf{Sd}] \leftarrow \text{NonDominateSort}(P)$  % 种群 $P$ 中Pareto前沿作为教师解集 $\mathbf{Te}$ , 其余解为学生解集 $\mathbf{Sd}$ 
2 for  $n = 1$  to  $\text{size}(\mathbf{Sd})$  do
3    $te = \text{SelRand}(\mathbf{Te}, 1)$  % 教师解集 $\mathbf{Te}$ 中随机选择一个教师解 $te$ 
4    $\rho = \text{DistAdpte}(\mathbf{Sd}(n), te, \mathbf{Te}, \rho_{\min}, \rho_{\max})$  % 计算学习强度
5    $\mathbf{Cte}(n) = \text{MPX}(\mathbf{Sd}(n), te, \rho)$  % 采用MPX算子实现学习, 得到子代学生解存入 $\mathbf{Cte}$ 
6 end for
7 for  $n = 1$  to  $\text{size}(\mathbf{Sd})$  do
8    $sd = \text{SelRand}(\mathbf{Sd}, 1)$  % 学生解集 $\mathbf{Sd}$ 中随机选择一个学生解 $sd$ 
9    $\rho = \text{DistAdpte}(\mathbf{Sd}(n), sd, \mathbf{Sd}, \rho_{\min}, \rho_{\max})$  % 计算学习强度
10   $\mathbf{Csd}(n) = \text{MPX}(\mathbf{Sd}(n), sd, \rho)$  % 采用MPX算子实现学习, 得到子代学生解存入 $\mathbf{Csd}$ 
11 end for
12  $C \leftarrow \text{MergeDiscardRep}(\mathbf{Cte}, \mathbf{Csd})$  % 将教授与学习阶段产生的子代解合并, 并去掉重复解
    
```

3.3 基于改进关键工序移动的邻域搜索

柔性作业车间调度析取图中存在最长不可压缩的工序序列, 这些工序被称为关键工序^[28]. 文献[29–30]等通过改变关键工序加工顺序和加工设备, 快速降低了完工时间. 如图5所示为本文采用的两种邻域结构, 用于更新OpSeq向量和OpAssign向量. 其中, 邻域结构1将一个关键工序移出其所在的连续关键工序块, 并更新执行器选择; 邻域结构2为一个关键工序选择位于其他设备本体的执行器, 并随机安排可行加工顺序.

值得注意的是, 传统的关键工序移动是完全随机的, 无法直接优化综合生产成本. 为实现完工时间和综合生产成本的平衡搜索, 本算法改进了关键工序及其执行器的选择机制: 先评估关键工序选择其他执行器的综合生产成本, 然后通过式(20)计算关键工序及执行器被选中的概率, 最后采用轮盘赌方式确定进行邻域搜索的关键工序及其执行器.

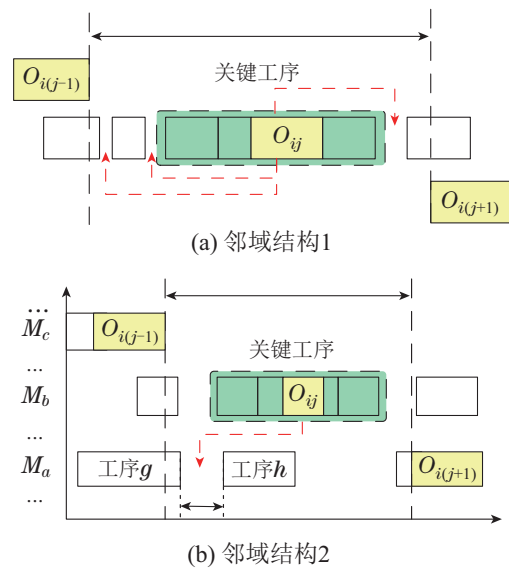


图5 基于关键工序移动的邻域结构

Fig. 5 Diagram of neighborhood structure based on critical operation movement

$$P_{rs} = \frac{CPC_{rs}}{\sum_{r \in \Theta} \sum_{s \in \pi} CPC_{rs}}, \quad (20)$$

式中: P_{rs} 表示关键工序 r 选择执行器 s 的概率; CPC_{rs} 表示关键工序 r 选择执行器 s 降低的综合生产成本(不考虑增加综合生产成本的选择); Θ 和 π 分别为当前关键工序集和可选执行器集. 该选择机制使关键工序移动更倾向于降低综合生产成本, 其效果将在后文验证.

4 仿真实验

COEAPS问题是标准柔性作业车间调度问题的拓展, 目前没有公开的标准测试集. 为此, 生成6个小规模算例(命名为S1-S6)和8个大规模测试算例(命名为L1-L8). 其中: S1-S6采用随机方式生成, L1-L8基于柔性作业车间调度问题的专用MK测试算例生成.

为了全面反映实际应用中的典型场景, 测试算例的参数设置覆盖了执行器数/设备本体数/插槽总容量受限的情况. COEAPS测试算例主要参数如表4所示, 表中 U 表示随机均匀分布, N 表示正态分布, T 为原MK测试算例的加工时间. 本文所用算例文件及相关说明可见于 github.com/modesty-twh/COEAPS.git.

选取世代距离 (generational distance, GD)、分散度 (Δ) 和反世代距离 (inverted GD, IGD) 分别评估前沿的收敛性、分布特性和综合性能, 指标的定义式参见文献[31], 指标值越小表明该项性能越优. 由于部分算例的真实前沿未知, 按照文献通用做法^[32-33], 将实验过程中找到的非支配解视为真实前沿. 经过预实验, DDTLO算法参数设置为: $n_{pop} = 50$, $\rho_{min} = 0.01$, $\rho_{max} = 0.3$. 精确求解通过Gurobi实现, 其他算法使用MATLAB R2020b编程, 运行在Intel Core i7(3.0 GHz), 8 GB内存的计算机上.

表4 COEAPS测试算例的参数

Table 4 Parameters of COEAPS benchmarks

名称	算例来源	工件数	工序数	执行器数	设备本体数	插槽总容量	加工时间	加工能耗
S1	—	2	5	5	2	3	$U(2, 22)$	$U(10, 30)$
S2	—	3	9	4	3	5	$U(2, 22)$	$U(10, 30)$
S3	—	3	9	10	3	5	$U(2, 22)$	$U(10, 30)$
S4	—	4	10	6	3	7	$U(2, 22)$	$U(10, 30)$
S5	—	4	13	10	4	7	$U(2, 22)$	$U(10, 30)$
S6	—	5	13	10	4	8	$U(2, 22)$	$U(10, 30)$
L1	MK06	10	50	10	9	17	$N(T, 0.2T)$	$U(10, 30)$
L2	MK08	30	285	10	8	21	$N(T, 0.2T)$	$U(10, 30)$
L3	MK09	20	240	10	7	13	$N(T, 0.2T)$	$U(10, 30)$
L4	MK10	20	240	15	6	12	$N(T, 0.2T)$	$U(10, 30)$
L5	MK12	30	193	10	5	8	$N(T, 0.2T)$	$U(10, 30)$
L6	MK13	30	231	10	10	18	$N(T, 0.2T)$	$U(10, 30)$
L7	MK14	30	277	15	7	14	$N(T, 0.2T)$	$U(10, 30)$
L8	MK15	30	284	15	6	14	$N(T, 0.2T)$	$U(10, 30)$

4.1 协同优化必要性验证

为验证COEAPS, 将执行器配置与生产调度进行协同优化的必要性, 以顺序优化(end-effector allocation and production scheduling, EAPS)作为对比. EAPS先将执行器随机配置到设备本体, 再优化工序分配和加工顺序. 为了降低偶然因素的干扰, 每个算例上均将执行器随机分配3次, 记为EAPS1-3. 实验在小规模算例S1-S6上开展. 如图6所示为COEAPS与EAPS1-3得到的Pareto前沿解的情况. 可以看出, COEAPS的Pareto前沿完全支配EAPS1-3的Pareto前沿. 因此, 将执行器配置与生产调度进行协同能够获得更加全局的优化结果.

4.2 教学算子和邻域搜索有效性验证

为了验证邻域搜索的有效性, 设计了两种DDTLO的变体作为对比, 分别命名为DDTLO1和DDTLO2.

其中: DDTLO1采用标准教学算子, DDTLO2不包含邻域搜索. 有效性实验在大规模算例L1-L8上开展, 各算法求解时间均为(工序数 \times 执行器数 \times 设备本体数)/100s. 评价指标结果如表5所示. 可以看出, DDTLO明显优于对比算法, 尤其在质量和综合性能指标上的优势非常显著. 同时注意到, DDTLO2的差异相对较大. 因此, DDTLO的优异性能主要得益于基于改进关键工序移动邻域搜索提供的强大局部搜索能力, 而基于距离自适应调整机制的教学算子进一步平衡了算法的搜索强度和种群多样性. 同时, 所提距离自适应调整机制不依赖问题结构, 可以应用于求解其他多目标优化问题, 具有广泛的适用性.

4.3 算法性能验证

DDTLO算法性能验证包括与Gurobi求解结果对比、与元启发式算法对比两个部分.

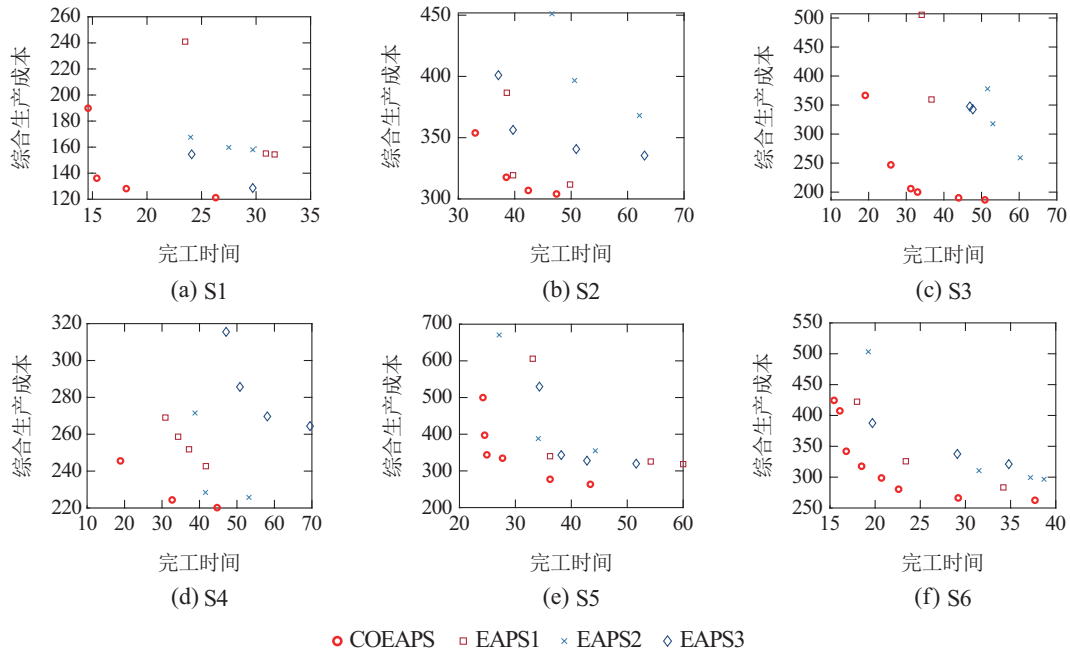


图 6 COEAPS和EAPS1-3的Pareto前沿对比

Fig. 6 Comparison of Pareto frontiers between COEAPS and EAPS1-3

表 5 DDTLO, DDTLO1和DDTLO2的评价指标结果

Table 5 Metric values obtained by DDTLO, DDTLO1 and DDTLO2

算例	GD			Δ			IGD		
	DDTLO	DDTLO1	DDTLO2	DDTLO	DDTLO1	DDTLO2	DDTLO	DDTLO1	DDTLO2
L1	0.012	0.021	0.032	0.947	0.966	0.976	0.010	0.026	0.034
L2	0.006	0.022	0.027	0.499	0.628	0.691	0.001	0.011	0.027
L3	0.011	0.041	0.043	0.963	0.951	0.98	0.004	0.015	0.024
L4	0.025	0.030	0.032	0.887	0.906	0.956	0.015	0.049	0.043
L5	0.012	0.023	0.037	0.968	0.956	0.987	0.013	0.029	0.039
L6	0.003	0.008	0.024	0.967	0.984	1.004	0.008	0.024	0.045
L7	0.001	0.017	0.033	1.005	0.993	0.978	0.010	0.013	0.023
L8	0.001	0.003	0.014	0.856	0.945	0.981	0.015	0.017	0.037

a) DDTLO与Gurobi求解结果对比.

实验在小规模算例S1-S6上进行, DDTLO求解时间为(工序数×执行器数×设备本体数)/10s, Gurobi求解每个前沿解的时间上限为7 200 s. 评价指标结果如表6所示. 在S1-S4算例上, DDTLO算法获得的解与精确解非常接近; 在S5和S6算例上, Gurobi难以在时限内找到精确解, 此时DDTLO算法获得的解优于Gurobi求解结果.

b) DDTLO与元启发式算法对比.

选择3种对比算法: NSGA-III(non-dominated sorting genetic algorithm)^[34], MOGWO(multi-objective grey wolf optimizer)^[35]和TLO-O(teaching learning metaheuristic algorithm with opposite-based learning)^[36]. 其中, TLO-O于2022年提出, 结合反向学习策略^[37]进行初始化, 并采用翻转、调换方式实现局部搜

索. 所有对比算法的种群规模均为50、求解时间均为(工序数×执行器数×设备本体数)/100 s. NSGA-III和MOGWO的交叉概率为0.1、变异概率为0.01、外部存储库规模为50. TLO-O的局部搜索概率为0.3.

表 6 DDTLO与Gurobi的评价指标结果

Table 6 Metric values obtained by DDTLO and Gurobi

算例	GD		Δ		IGD	
	DDTLO	Gurobi	DDTLO	Gurobi	DDTLO	Gurobi
S1	0	0	0	0	0	0
S2	0	0	0	0	0	0
S3	0.001	0	0.089	0	0.006	0
S4	0.005	0.003	0.078	0	0.007	0.001
S5	0.002	0.094	0.031	0.981	0.018	0.214
S6	0.001	0.021	0.014	1.003	0.004	0.036

评价指标结果如表7所示. 可以看出, 在收敛性和分散性方面, DDTLO在7/8的算例中获得了最优值,

而在综合性能方面, DDTLO均显著优于对比算法. 因此, DDTLO在解决COEAPS问题上具有较强竞争力.

表7 DDTLO与元启发式算法对比的评价指标结果
Table 7 Metric values obtained by DDTLO and comparative meta-heuristics

算例	GD				Δ				IGD			
	DDTLO	NSGA-III	MOGWO	TLO-O	DDTLO	NSGA-III	MOGWO	TLO-O	DDTLO	NSGA-III	MOGWO	TLO-O
L1	0.008	0.046	0.045	0.006	0.972	1.012	0.98	1.019	0.016	0.051	0.029	0.039
L2	0.001	0.006	0.009	0.027	0.646	1.014	0.979	1.09	0.001	0.024	0.022	0.05
L3	0.004	0.006	0.121	0.02	0.94	1.011	0.996	1.141	0.002	0.034	0.112	0.019
L4	0.001	0.02	0.018	0.009	0.935	1.078	0.988	1.335	0.012	0.022	0.044	0.022
L5	0.004	0.006	0.008	0.006	0.869	1.009	0.946	1.021	0.012	0.036	0.027	0.028
L6	0.004	0.019	0.027	0.022	0.985	1.004	0.974	1.026	0.009	0.052	0.096	0.035
L7	0.01	0.031	0.041	0.019	0.987	1.009	0.994	1.004	0.006	0.02	0.062	0.029
L8	0.005	0.025	0.033	0.011	0.945	1.020	0.996	1.049	0.005	0.032	0.054	0.013

5 结论

执行器配置是柔性作业的重要决策问题. 本文以降低综合生产成本和缩短完工时间为目标, 提出一种基于改进教学优化算法的执行器配置与生产调度协同优化方法. 结果表明, 本文所提协同优化方法能够提高决策的全局性, 从而获得更优加工规划方案; 所提DDTLO算法在与精确求解器和同类算法的对比中, 表现出优异的求解性能和优势. 本文方法紧密贴合实际加工场景, 为柔性作业车间的智能化决策提供了必要的建模和求解工具. 未来的研究可进一步推广到分布制造环境下, 并考虑执行器调度、完成时间约束等实际情况.

参考文献:

- [1] LI Xinyu, LI Zhaofu, GAO Liang. Paths for the digital transformation and intelligent upgrade of China's discrete manufacturing industry. *Strategic Study of CAE*, 2022, 24(2): 63 – 74. (李新宇, 李昭甫, 高亮. 离散制造行业数字化转型与智能化升级路径研究. 中国工程科学, 2022, 24(2): 63 – 74.)
- [2] LU Bingheng, SHAO Xinyu, ZHANG Jun, et al. Development strategy for intelligent factory in discrete manufacturing. *Strategic Study of CAE*, 2018, 20(4): 44 – 50. (卢秉恒, 邵新宇, 张俊, 等. 离散型制造智能工厂发展战略. 中国工程科学, 2018, 20(4): 44 – 50.)
- [3] SONG Simeng, JIANG Zengqiang, MA Jing, et al. Operation mechanism and reconfiguration methods of modular production systems based on digital twin. *Computer Integrated Manufacturing Systems*, 2021, 27(2): 510 – 520. (宋思蒙, 蒋增强, 马靖, 等. 基于数字孪生的模块化生产系统运行机制及重构方法. 计算机集成制造系统, 2021, 27(2): 510 – 520.)
- [4] ZHOU Guanghui, FU Xiangjing. Multi-objective scheduling model of limited tools in low-carbon manufacturing environment. *Journal of Xi'an Jiaotong University*, 2017, 51(10): 7 – 13. (周光辉, 傅祥璟. 低碳制造环境下车间有限刀具多目标调度研究. 西安交通大学学报, 2017, 51(10): 7 – 13.)
- [5] LI L, LI C, TANG Y, et al. An integrated solution to minimize the energy consumption of a resource-constrained machining system. *IEEE*

Transactions on Automation Science and Engineering, 2020, 17(3): 1158 – 1175.

- [6] AN Y J, CHEN X H, ZHANG J, et al. A hybrid multi-objective evolutionary algorithm to integrate optimization of the production scheduling and imperfect cutting tool maintenance considering total energy consumption. *Journal of Cleaner Production*, 2020, 268: 121540.
- [7] ZHANG L K, DENG Q W, GONG G L, et al. A new unrelated parallel machine scheduling problem with tool changes to minimise the total energy consumption. *International Journal of Production Research*, 2020, 58(22): 6826 – 6845.
- [8] DANG Q V, DIESEN T V, MARTAGAN T, et al. A matheuristic for parallel machine scheduling with tool replacements. *European Journal of Operational Research*, 2021, 291(2): 640 – 660.
- [9] ZHU Yingying, WU Zhengjia, TANG Qihua, et al. Bi-objective parallel machine equally lot-sizing and scheduling problem under limited number of sublots. *Computer Integrated Manufacturing Systems*, 2022, 28(11): 3379 – 3390. (朱颖颖, 吴正佳, 唐秋华, 等. 批次受限的双目标并行机等量分批调度. 计算机集成制造系统, 2022, 28(11): 3379 – 3390.)
- [10] WANG S L, ZOU H X, WANG S B. A tabu-GA-based parallel machine scheduling with restrained tool resources. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 2022, 236(1/2): 39 – 50.
- [11] ZHENG X L, WANG L. A two-stage adaptive fruit fly optimization algorithm for unrelated parallel machine scheduling problem with additional resource constraints. *Expert Systems with Applications*, 2016, 65: 28 – 39.
- [12] RAO R V, SAVSANI V J, VAKHARIA D P. Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. *CAD Computer Aided Design*, 2011, 43(3): 303 – 315.
- [13] SONG Qiang. A hybrid multi-objective teaching-learning-based optimization algorithm for unrelated parallel machine scheduling problem. *Control Theory & Applications*, 2020, 37(10): 2242 – 2256. (宋强. 求解异构并行机调度问题的混合多目标教学优化算法. 控制理论与应用, 2020, 37(10): 2242 – 2256.)
- [14] LI Rui, GONG Wenyin. An improved multi-objective evolutionary algorithm based on decomposition for bi-objective fuzzy flexible job-shop scheduling problem. *Control Theory & Applications*, 2022, 39(1): 31 – 40. (李瑞, 龚文引. 改进的基于分解的多目标进化算法求解双目标模糊柔性作业车间调度问题. 控制理论与应用, 2022, 39(1): 31 – 40.)

- [15] ZHOU Yongqiang, WANG Cuiyu, LI Yingli, et al. Improved fruit fly optimization algorithm for solving the hybrid flow shop scheduling problem. *Control Theory & Applications*, 2023, 40(4): 597 – 606. (周永强, 王翠雨, 李颖俐, 等. 改进果蝇算法求解混合流水车间调度问题. *控制理论与应用*, 2023, 40(4): 597 – 606.)
- [16] LI Ming, LEI Deming. Novel imperialist competitive algorithm for many-objective flexible job shop scheduling. *Control Theory & Applications*, 2019, 36(6): 893 – 901. (李明, 雷德明. 基于新型帝国竞争算法的高维多目标柔性作业车间调度. *控制理论与应用*, 2019, 36(6): 893 – 901.)
- [17] LI Junqing, LI Wenhan, TAO Xinrui, et al. A survey on time constrained hybrid flow shop scheduling problems. *Control Theory & Applications*, 2020, 37(11): 2273 – 2290. (李俊青, 李文涵, 陶昕瑞, 等. 时间约束混合流水车间调度问题综述. *控制理论与应用*, 2020, 37(11): 2273 – 2290.)
- [18] TAN W H, YUAN X F, WANG J L, et al. Multi-objective teaching-learning-based optimization algorithm for carbon-efficient integrated scheduling of distributed production and distribution considering shared transportation resource. *Journal of Cleaner Production*, 2023, 406: 137061.
- [19] TANG H, LEE J. Chaotic enhanced teaching-based differential evolution algorithm applied to discrete truss optimization. *Structures*, 2023, 49: 730 – 747.
- [20] SHAO W, PI D C, SHAO Z S. An extended teaching-learning based optimization algorithm for solving no-wait flow shop scheduling problem. *Applied Soft Computing Journal*, 2017, 61: 193 – 210.
- [21] ASHRAF M M, MALIK T N. A hybrid teaching-learning-based optimizer with novel radix-5 mapping procedure for minimum cost power generation planning considering renewable energy sources and reducing emission. *Electrical Engineering*, 2020, 102(4): 2567 – 2582.
- [22] LIU Q, ZHAN M, CHEKEM F O, et al. A hybrid fruit fly algorithm for solving flexible job-shop scheduling to reduce manufacturing carbon footprint. *Journal of Cleaner Production*, 2017, 168: 668 – 678.
- [23] WANG Y, CHANG M, HUANG X, et al. Cutting tool wear prediction based on the multi-stage Wiener process. *The International Journal of Advanced Manufacturing Technology*, 2023, 129(11/12): 5319 – 5333.
- [24] GAO T, ZHU H, WU J, et al. Hybrid physics data-driven model-based fusion framework for machining tool wear prediction. *The International Journal of Advanced Manufacturing Technology*, 2024, 132(3/4): 1481 – 1496.
- [25] 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.
- [26] ZHENG Xiaocao, GONG Wenjin, et al. An improved artificial bee colony algorithm for fuzzy flexible job-shop scheduling problem. *Control Theory & Applications*, 2020, 37(6): 1284 – 1292. (郑小操, 龚文引. 改进人工蜂群算法求解模糊柔性作业车间调度问题. *控制理论与应用*, 2020, 37(6): 1284 – 1292.)
- [27] LUO Wenchong, QIAN Bin, HU Rong, et al. Hyper-heuristic cross-entropy algorithm for distributed assembly flexible job-shop scheduling problem. *Control Theory & Applications*, 2021, 38(10): 1551 – 1568. (罗文冲, 钱斌, 胡蓉, 等. 超启发式交叉熵算法求解分布式装配柔性作业车间调度问题. *控制理论与应用*, 2021, 38(10): 1551 – 1568.)
- [28] MASTROLILLI M, GAMBARDELLA L M. Effective neighbourhood functions for the flexible job shop problem. *Journal of Scheduling*, 2000, 3(1): 3 – 20.
- [29] ZHAO F, HE X, WANG L. A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem. *IEEE Transactions on Cybernetics*, 2021, 51(11): 5291 – 5303.
- [30] HE X, PAN Q K, GAO L, et al. A greedy cooperative co-evolutionary algorithm with problem-specific knowledge for multiobjective flow-shop group scheduling problems. *IEEE Transactions on Evolutionary Computation*, 2023, 27(3): 430 – 444.
- [31] TAN W H, YUAN X F, HUANG G M, et al. Low-carbon joint scheduling in flexible open-shop environment with constrained automatic guided vehicle by multi-objective particle swarm optimization. *Applied Soft Computing*, 2021, 111: 107695.
- [32] LUO S, ZHANG L X, FAN Y X. Energy-efficient scheduling for multi-objective flexible job shops with variable processing speeds by grey wolf optimization. *Journal of Cleaner Production*, 2019, 234: 1365 – 1384.
- [33] LU C, LI X Y, GAO L, et al. An effective multi-objective discrete virus optimization algorithm for flexible job-shop scheduling problem with controllable processing times. *Computers and Industrial Engineering*, 2017, 104: 156 – 174.
- [34] 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.
- [35] MIRJALILI S, SAREMI S, MIRJALILI S M, et al. Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization. *Expert Systems with Applications*, 2016, 47: 106 – 119.
- [36] BALANDE U, SHRIMANKAR D. A modified teaching learning metaheuristic algorithm with opposite-based learning for permutation flow-shop scheduling problem. *Evolutionary Intelligence*, 2022, 15(1): 57 – 79.
- [37] RAHNAMAYAN S, TIZHOOSH H R, SALAMA M M A. Opposition-based differential evolution. *IEEE Transactions on Evolutionary Computation*, 2008, 12(1): 64 – 79.

作者简介:

谭伟华 讲师, 博士, 目前研究方向为离散制造系统仿真与优化决策, E-mail: tanwh@hnust.edu.cn;

吴亮红 教授, 博士生导师, 博士, 目前研究方向为群智能系统控制理论, E-mail: lhwu@hnust.edu.cn;

李哲 副教授, 博士生导师, 博士, 目前研究方向为信息物理系统, E-mail: zheli@hnu.edu.cn;

袁小芳 教授, 博士生导师, 博士, 目前研究方向为智能系统优化理论与应用, E-mail: yuanxiaofang@hnu.edu.cn.