

3-22-2023

Hyper-heuristic Three Dimensional EDA for Solving Green Two-Sided Assembly Line Balancing Problem

Rong Hu

1.School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China;, ronghu@vip.163.com

Shuai Ding

1.School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China;

Bin Qian

1.School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China;2.School of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China;

Changsheng Zhang

1.School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Hyper-heuristic Three Dimensional EDA for Solving Green Two-Sided Assembly Line Balancing Problem

Abstract

Abstract: This paper establishes a model for green robotic two-sided assembly line balancing problem of type-I, and a hyper-heuristic three dimensional estimation of distribution algorithm (HH3DEDA) is proposed for solving this problem. In HH3DEDA, a combinatorial encoding rule based on process selectors is designed via considering the characteristics of the problem. Then, HH3DEDA with a high and low layered structure is proposed. In the upper layer, the three-dimensional probability matrix is utilized to learn high-quality high individual block structure and its distribution information, and then the matrix is sampled to generate new high level individuals. Each high individual is composed by 12 heuristic operations, which are designed via considering the characteristics of the problem. In the lower layer, the high individual determined heuristic operation permutation is used as a new heuristic to perform a deep search for the GRTALBP-I solution space. Meanwhile, the energy saving strategy of robot switching machine is utilized to enhance the quality of obtaining the non-dominated individuals. Simulation experiments demonstrate the effectiveness of the proposed algorithm.

Keywords

two-sided assembly balancing, hyper-heuristic, three-dimensional distribution estimation algorithm, multi-objective optimization, saving energy and reducing consumption

Recommended Citation

Rong Hu, Shuai Ding, Bin Qian, Changsheng Zhang. Hyper-heuristic Three Dimensional EDA for Solving Green Two-Sided Assembly Line Balancing Problem[J]. Journal of System Simulation, 2023, 35(3): 454-469.

超启发式三维 EDA 求解绿色双边装配线平衡问题

胡蓉¹, 丁帅¹, 钱斌^{1,2}, 张长胜¹

(1. 昆明理工大学 信息工程与自动化学院, 云南 昆明 650500; 2. 昆明理工大学 机电工程学院, 云南 昆明 650500)

摘要: 针对绿色机器人的第 I 类双边装配线平衡问题(green robotic two-sided assembly line balancing problem of type-I, GRTALBP-I), 建立问题模型并提出一种超启发式三维分布估计算法(hyper-heuristic three dimensional estimation of distribution algorithm, HH3DEDA)进行求解。在 HH3DEDA 中, 结合问题特征, 设计基于工序选择因子的组合编码, 进而设计高低分层结构的 HH3DEDA。在高层, 采用三维概率矩阵学习优质高层个体中块结构及其分布信息, 后通过采样该矩阵以生成新的高层个体, 其中高层个体由结合问题特点设计的 12 种启发式操作的排列构成; 在低层, 将高层每个个体所确定启发式操作排列作为一种新的启发式算法对 GRTALBP-I 解空间执行较深入搜索。同时, 引入机器人开关机节能策略, 进一步提升所获取非支配解的质量。通过仿真对比实验, 验证了所提算法的有效性。

关键词: 双边装配线平衡; 超启发算法; 三维分布估计算法; 多目标优化; 节能降耗

中图分类号: TP391.9 文献标志码: A 文章编号: 1004-731X(2023)03-0454-16

DOI: 10.16182/j.issn1004731x.joss.21-1134

引用格式: 胡蓉, 丁帅, 钱斌, 等. 超启发式三维 EDA 求解绿色双边装配线平衡问题[J]. 系统仿真学报, 2023, 35(3): 454-469.

Reference format: Hu Rong, Ding Shuai, Qian Bin, et al. Hyper-heuristic Three Dimensional EDA for Solving Green Two-Sided Assembly Line Balancing Problem[J]. Journal of System Simulation, 2023, 35(3): 454-469.

Hyper-heuristic Three Dimensional EDA for Solving Green Two-Sided Assembly Line Balancing Problem

Hu Rong¹, Ding Shuai¹, Qian Bin^{1,2}, Zhang Changsheng¹

(1. School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China;

2. School of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China)

Abstract: This paper establishes a model for green robotic two-sided assembly line balancing problem of type-I, and a hyper-heuristic three dimensional estimation of distribution algorithm (HH3DEDA) is proposed for solving this problem. In HH3DEDA, a combinatorial encoding rule based on process selectors is designed via considering the characteristics of the problem. Then, HH3DEDA with a high and low layered structure is proposed. In the upper layer, the three-dimensional probability matrix is utilized to learn high-quality high individual block structure and its distribution information, and then the matrix is sampled to generate new high level individuals. Each high individual is composed by 12 heuristic operations, which are designed via considering the characteristics of the problem. In the lower layer, the high individual determined heuristic operation permutation is used as a new heuristic to perform a deep search for the GRTALBP-I solution space. Meanwhile, the energy saving strategy of robot switching machine is utilized to enhance the quality of obtaining the non-dominated individuals. Simulation

收稿日期: 2021-11-05 修回日期: 2022-02-09

基金项目: 国家自然科学基金(61963022, 62173169)

第一作者: 胡蓉(1974-), 女, 副教授, 硕士, 研究方向为智能优化调度、物流优化。E-mail: ronghu@vip.163.com

experiments demonstrate the effectiveness of the proposed algorithm.

Keywords: two-sided assembly balancing; hyper-heuristic; three-dimensional distribution estimation algorithm; multi-objective optimization; saving energy and reducing consumption

0 引言

装配线平衡问题(assembly line balancing problem, ALBP)广泛存在于实际生产中。该问题于1955年由Salveson^[1]提出,主要研究如何合理地对待作业工序分配至不同工位,以提升装配线的生产效率。20世纪90年代以来,双边装配线由于在设备利用率、生产效率等方面具有显著优势^[2],开始在汽车制造公司得以大量应用^[3-4],并吸引了学术界的关注。Bartholdi^[5]提出了双边装配线平衡问题(two-sided assembly line balancing problem, TALBP),并证明其属于NP难问题。近年来,国内外学者对TABLP已展开了较广泛的研究。按照已知变量与优化目标的不同,可将其分为以下3类问题:①已知节拍时间,以最小化工位数为优化目标的第一类双边装配线平衡问题(TABLP-I)^[6-8];②已知工位数量,以最小化节拍时间为优化目标的第二类双边装配线平衡问题(TABLP-II)^[9];③已知工位数量和节拍时间,以最大化装配效率为优化目标的第三类(第E类)双边装配线平衡问题(TABLP-III)^[10]。

然而,随着近年智能制造和人工智能的高速发展,工业机器人替代人工已成为构建智能化工厂的重要手段。因此,研究并求解机器人双边装配线平衡问题(robot TALBP, RTALBP)具有重要的现实意义。目前RTALBP已得到一定的研究。譬如,在单目标优化方面,文献[11]针对以最小化节拍时间为优化目标的RTALBP-II,设计了一种离散布谷鸟搜索算法(discrete cuckoo search algorithm, DCSA)进行求解。文献[12]针对以最小化节拍时间为优化目标的带序相关设置时间的RTALBP-II,建立问题模型并设计了一种混合元启发式算法(hybrid meta-heuristic algorithm, HMHA)进行求解。在多目标优化方面,文献[13]

针对以最小化工位数量、最大化平滑指数以及最大化生产线利润为优化目标的随机型双边装配线平衡问题(stochastic TALBP, STALBP),设计了一种离散花粉算法(discrete flower pollination algorithm, DLPA)进行求解。文献[14]针对以最小化周期时间和能耗为优化目标的RTALBP-II,设计了一种重启模拟退火算法(restarted simulated annealing algorithm, RSA)进行求解。根据上述文献,尚无考虑机器人节能优化策略的第I类RTALBP(以下简称RTALBP-I)平衡问题。故本文结合第I类双边装配线平衡问题(TALBP-I)与机器人双边装配线第II类平衡问题(RTALBP-II),并进一步考虑在装配过程中机器人开关机节能策略,提出以同时最小化考虑线效率及平滑指数的综合指标和总能耗指标为优化目标的绿色机器人第I类双边装配线平衡问题(green RTALBP-I, GRTALBP-I)模型。由于GRTALBP-I可归约为RTALBP-I,而RTALBP-I又可归约为TALBP,故GRTALBP-I也属于NP-hard问题。这表明研究GRTALBP-I也具有重要的理论意义。

超启发式算法(hyper-heuristic algorithm, HHA)是一类新型智能优化算法,该算法通过某种高层策略(high-level strategy, HLS)来操纵低层启发式算法(low-level heuristic, LLH),进而实现对解空间不同区域的搜索。近年来,HHA已成功应用于各类组合优化问题上^[15-17]。HHA的高层解空间类似于大峡谷的地貌特征,众多深浅各异的山谷(即LLHs构成的优质排列)存在于其庞大而扁平的解空间内,优质排列之间存在着相似的结构信息。同时,LLHs通过动态混合构成多种不同的启发式算法,可在算法搜索到达问题解空间中多种邻域结构的局部最优非支配解前保持一直向下搜索的状态。因此,设计有效的HLS与LLHs是提升算

法效率的关键。故本文设计超启发式三维分布估计算法进行求解,该算法高层采用三维概率模型优化高层操作序,能有效避免传统进化算法中普遍存在的对较优解模式破坏的问题^[18]。同时,三维概率模型较二维概率模型而言,可以同时学习和积累高层优质个体中相似块的序信息和位置信息,继而在采样生成新个体时能较准确地确定优质块结构在解中的具体位置,从而可更合理地引导算法搜索方向以提高搜索效率。此外,由相关文献可知,HH3尚未在装配线平衡这类问题中得到应用。

综上,本文研究绿色机器人第I类双边装配线平衡问题的建模与求解。在建模方面,建立以同时最小化线效率及平滑指数的综合指标与总能耗指标为优化目标的问题模型。在求解方面,结合GRTALBP-I的特点,设计超启发式三维分布估计算法(hyper-heuristic three-dimensional estimation of distribution algorithm, HH3DEDA)进行求解。设计新颖的工序、机器人两段式编解码机制,提出一种改进的NEH启发式算法作为种群初始化策略来确保初始种群的质量,在所提算法的高层采用三维分布估计算法(estimation of distribution algorithm, EDA)来合理学习和积累优质高层个体中块结构及其分布信息,其中各高层个体由所设

计的12种启发式操作的某种排列构成;同时低层将高层个体中的启发式操作按顺序执行,从而实现了对问题解空间不同区域的较深入搜索。此外,设计机器人开关机节能策略,进一步提升算法所获取非支配解的质量。通过仿真对比实验,验证了HH3DEDA的有效性。

1 GRTALBP-I问题描述

在双边装配线中,将一对面对面的左右工位称为成对工位,左右工位上分配不同类型的工序并同时进行装配。因此,相比于直线型装配线,在相同节拍时间和工序数量下,双边装配线缩短了装配线的长度,提高了设备的利用率,工位的实际操作时间也更接近于节拍时间,故双边装配线具备更高的生产效率和更少的设备投资。

现以上海某汽车工厂的汽车装配线为例对GRTALBP-I进行描述。该装配线为双边布局,具有386台高精度机器人,不同工位按照实际需求分配不同类型的机器人。工厂内总装车间、车身车间、变速箱车间、电池装配车间等均采用机器人自主作业,其机器人可以高效率作业自冲铆接、电阻焊接、外总拼侧围涂胶和外总拼焊接等工序,且工厂内机器人双边装配线布局如图1所示。

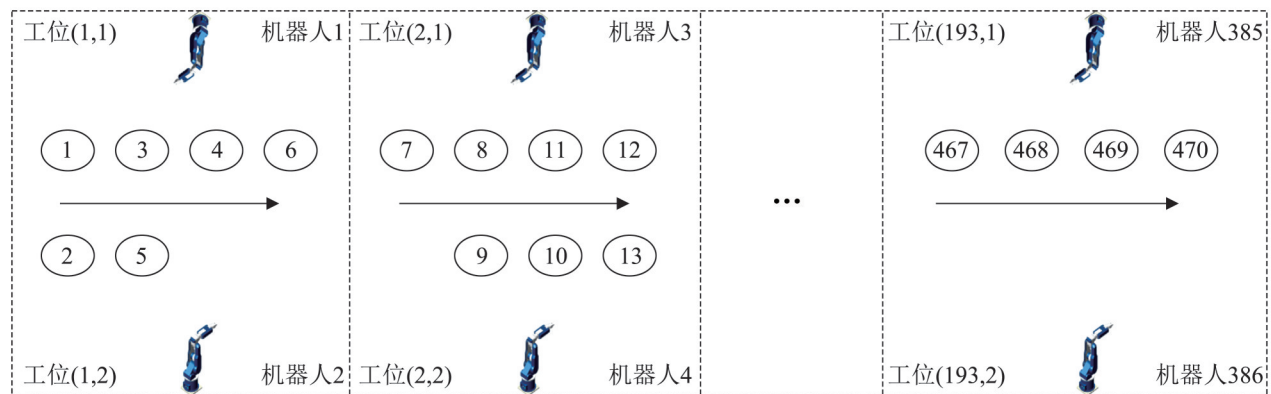


图1 机器人双边装配线布局
Fig. 1 Robot bilateral assembly line layout

图2为该工厂内部分工序优先约束关系图, 其中圆圈内的数字为工序序号, 括号内的数字为工序单位作业时间, 字母为该工序的可分配方向: L表示该工序可分配到左侧工位, R表示该工序可分配到右侧工位, E表示该工序可分配到左右任意一侧工位, 箭头为工序之间的优先约束关系。

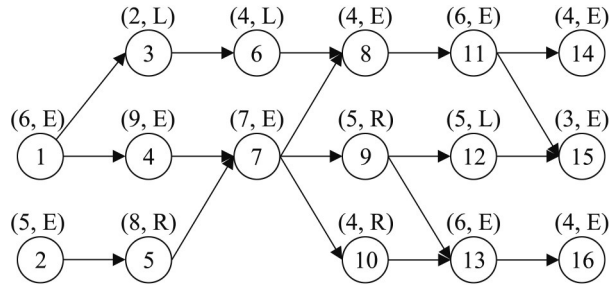


图2 工序优先约束关系图

Fig. 2 Priority constraint relationship between process

为使GRTALBP-I更贴合实际生产环境, 本文在优先顺序约束、节拍约束、操作方位约束等基本约束的基础上, 还考虑了双边装配线中所特有的其他约束, 如位置约束、区域约束和同步约束等^[19]。其中, 位置约束指有些工序需要借助特定的设备进行装配, 而这些设备一般在固定的工位上, 因此这部分工序的工位是确定的。区域约束分为积极区域约束和消极区域约束: 积极区域约束指有些工序在装配中需要相同的工具而分配到相同的工位上; 消极区域约束指出于安全因素考虑有些工序不能配到相同成对工位上。同步约束指有些工序必须在两侧工位上同时开始装配。

1.1 符号定义

GRTALBP-I所涉及到的有关符号定义如表1所示。

1.2 问题假设

对GRTALBP-I做出如下假设:

(1) 假设每种类型机器人的能耗不同, 总能耗通过每个机器人的具体能耗来计算。

(2) 假设装配线只组装单一产品。

表1 符号定义表

Table 1 Symbol definition

符号	定义
I	工序集合 $I = \{1, 2, \dots, Nt\}$, $i, l, p \in I$
J	成对工位集合 $J = \{1, 2, \dots, Nm\}$, $j, h \in J$
R	机器人类型集合 $R = \{1, 2, \dots, m\}$, $m \in R$ 其中 r 为类型 m 的机器人
Nt	工序数量
Nm	成对的工位数量
NR_r	机器人 r 的可用数量
NR	可用机器人数量
t_{i0}	工序 i 的作业时间, $i \in I$
t_{ir}	工序 i 使用机器人 r 的作业时间, $i \in I, r \in R$
t_i^f	工序 i 的完工时间
CT	节拍时间
t_{off_r}	机器人 r 关机所需时间
t_{on_r}	机器人 r 开机所需时间
wt_{ir}	工序 i 完工后机器人 r 的空载时间
WT_{ir}	满足条件的 wt_{ir} 构成的集合
T_{min_r}	机器人 r 空载时, 执行开关机所需的最小时间
N_r	机器人 r 可执行开关机次数
EC_r	机器人 r 每单位时间的能耗
SEC_r	机器人 r 每单位时间的待机能耗
EC_{off-on}	机器人 r 执行开关机所需能耗
EC_{save}	机器人 r 执行开关机所节省的能耗
EC_{jk}	成对工位 j 的 k 侧的能耗
TEC	总能耗
$K(i)$	工序 i 优先装配方向集合
$C(i)$	工序 i 装配方向相反的集合
P_0	没有直接前序任务的工序集合
$P(i)$	工序 i 的直接前序任务集合
$P_a(i)$	工序 i 的所有前序任务集合
$S_a(i)$	工序 i 的所有后序任务集合
$S(i)$	工序 i 的直接后序任务集合
μ	足够大的正整数
PC	位置约束的工序集合
PZ	积极区域约束的工序集合
NZ	消极区域约束的工序集合
SC	同步约束的工序集合
x_{ijk}	如果工序 i 分配到工位 (j, k) , 则取值1; 否则为0
y_{rjk}	如果机器人 r 分配到工位 (j, k) , 则取值1; 否则为0
z_{ip}	如果在同一个工位内工序 i 早于工序 p 分配, 取值1; 否则取0
α_{ijk}	如果 $\alpha_{ijk} = 1$, 机器人 r 关机; 否则, 保持开机

(3) 假设工序的作业时间取决于分配的机器人类型，且作业时间是不确定的。

(4) 假设每个工作站分配一个机器人，且工作站的数量等于所用机器人的数量。

(5) 假设每个工序均可被任一机器人作业，每个机器人可以被分配到任何工作站；但同一工序不可被不同机器人同时作业。

(6) 假设不考虑工序之间的设置时间以及设备维护操作。

1.3 数学模型

本文在双边装配线第 I 类平衡问题模型^[8]和机器人双边装配线第 II 平衡问题模型^[14]基础上，同时考虑经济指标与能耗指标的平衡，提出了绿色机器人的第 I 类双边装配线平衡问题(GRTALBP-I)模型。在 GRTALBP-I 中，采用机器人开关机节能策略来平衡装配过程中的能耗指标，具体描述如下：

确保每个工序只能分配到一个工作站：

$$\sum_{j \in J} \sum_{k \in K(i)} x_{ijk} = 1, \forall i \in I \quad (1)$$

节拍时间约束为

$$t_i^f \leq CT, \forall i \in I \quad (2)$$

式(3)~(6)为优先顺序约束。对于一对工序 (i, l) ，若 l 为 i 的直接前序工序，且 2 个工序分配到相同成对工位 j 中，则适用于式(4)；若 2 个工序没有优先约束关系，则适用于式(5)和(6)：

$$\sum_{h \in J} \sum_{k \in K(l)} h \cdot x_{ijk} \leq \sum_{j \in J} \sum_{k \in K(i)} j \cdot x_{ijk}, \forall i \in I - P_0, h \in P(i) \quad (3)$$

$$t_i^f - t_l^f + \mu(1 - \sum_{k \in K(l)} x_{ljk}) + \mu(1 - \sum_{k \in K(i)} x_{ijk}) \geq \sum_{r=1}^m \sum_{r=1}^{NR_r} t_{ir} \cdot y_{rjk}, \forall i \in I - P_0, h \in P(i) \quad (4)$$

$$t_p^f - t_i^f + \mu(1 - x_{ijk}) + \mu(1 - x_{pjk}) + \mu(1 - z_{ip}) \geq \sum_{r=1}^m \sum_{r=1}^{NR_r} t_{pr} \cdot y_{rjk}, \forall i \in I, p \in I - (P_a(i) \cup S_a(i) \cup C(i)) \text{ 且 } p > i, j \in J, k \in K(i) \cap K(p) \quad (5)$$

$$t_i^f - t_p^f + \mu(1 - x_{ijk}) + \mu(1 - x_{pjk}) + \mu \cdot z_{ip} \geq \sum_{r=1}^m \sum_{r=1}^{NR_r} t_{ir} \cdot y_{rjk},$$

$$\forall i \in I, p \in I - (P_a(i) \cup S_a(i) \cup C(i)) \text{ 且 } p > i, j \in J, k \in K(i) \cap K(p) \quad (6)$$

式(7)确保工序 i 的完工时间大于或等于操作：

$$t_i^f + \mu(1 - x_{ijk}) \geq \sum_{r=1}^m \sum_{r=1}^{NR_r} t_{ir} \cdot y_{rjk}, \forall i \in I, j \in J, k \in K(i) \quad (7)$$

式(8)，(9)确保每个工作站只能分配一台机器人：

$$\sum_{r=1}^m \sum_{r=1}^{NR_r} y_{rjk} = 1, \forall j \in J, k = 1, 2 \quad (8)$$

$$\sum_{j=1}^{Nm} \sum_{k=1}^2 y_{rjk} = 1, \forall m \in R, r \leq NR \quad (9)$$

式(10)~(12)分别为位置约束、积极区域约束和消极区域约束：

$$x_{ijk} = 1, \forall (i, j, k) \in PC, k \in K(i) \quad (10)$$

$$x_{ijk} - x_{ljk} = 0, \forall (i, l) \in PZ, k \in K(i) \cap K(l) \quad (11)$$

$$\sum_{k \in K(i)} x_{ijk} + \sum_{k \in K(l)} x_{ljk} \leq 1, \forall (i, l) \in NZ \quad (12)$$

式(13)，(14)为同步约束：

$$x_{ijr} - x_{ljk} = 0, \forall (i, l) \in SC, k \in K(l), f \in K(i), f \neq k \quad (13)$$

$$t_i^f - t_i = t_l^f - t_l, \forall (i, l) \in SC \quad (14)$$

式(15)和式(16)为机器人执行开关机的条件：

$$T_{Mr} = (EC_{\text{off-on}} / SEC_r) \vee (t_{\text{off}} + t_{\text{on}}) \quad (15)$$

$$\alpha_{rjk} = \begin{cases} 1, & \text{若 } wt_{ir} \geq T_{Mr} \text{ 且 } wt_{ir} \in WT_{ir} \\ 0, & \text{否则} \end{cases} \quad (16)$$

式(17)，(18)为成对工作站 j 的 k 侧的能耗计算公式：

$$EC_{\text{save}} = (\sum_{r=1}^m \sum_{r=1}^{NR_r} SEC_r \cdot wt_{ir} \cdot y_{rjk} - EC_{\text{off-on}}) \cdot \alpha_{rjk} \quad (17)$$

$$EC_{jk} = \sum_{r=1}^m \sum_{r=1}^{NR_r} \sum_{i=1}^{N_i} EC_r \cdot t_{ir} \cdot x_{ijk} \cdot y_{rjk} + (\sum_{r=1}^m \sum_{r=1}^{NR_r} SEC_r \cdot y_{rjk}) \times (CT - \sum_{r=1}^m \sum_{r=1}^{NR_r} \sum_{i=1}^{N_i} t_{ir} \cdot x_{ijk} \cdot y_{rjk}) - EC_{\text{save}} \quad (18)$$

1.3.1 目标函数

本文以同时最小化线效率(LE)及平滑指数(SI)

的综合指标与总能耗(TEC)为优化目标。第一个优化目标(即综合指标)中的LE在求解过程中要求实现最大化以降低装配线的长度并减少工位数量, SI则要求实现最小化以降低不同工作站之间的负载差异, 同时引入LE和SI这2个指标将更符合实际生产需求^[20]。LE和SI的计算公式为

$$Max = \max \left\{ \left[\frac{L_{Total}}{CT} \right], \left[\frac{R_{Total}}{CT} \right] \right\} \quad (19)$$

$$LB = 2 \cdot Max + \max \left\{ 0, \left[\frac{E_{Total} - (Max \cdot CT - L_{Total}) - (Max \cdot CT - R_{Total})}{CT} \right] \right\} \quad (20)$$

$$LE = \frac{LB}{m_R + m_L} \times 100 \quad (21)$$

$$SI = \frac{\sqrt{\sum_w^{m_R} (S_{max} - S_w)^2 + \sum_q^{m_L} (S_{max} - S_q)^2}}{\sqrt{(m_R + m_L)}} \quad (22)$$

式中: L_{Total} , R_{Total} , E_{Total} 分别为L、R、E类工序的总时间; S_{max} 为最大的工位作业时间; S_w 和 S_q 分别为左、右边第 w 个工位和第 q 个工位的作业时间; m_R 和 m_L 分别为左、右边工位数量。

实验表明, LE和SI这2个目标并不是互相冲突的关系, 且LE在到达最优解后不再变化, 而SI依然朝着最优解方向进化, 故最终只会生成1个Pareto最优解^[20]。因此, 将这2个目标合并为1个目标后同时优化, 将获得更有意义的分配方案。合并后的公式为

$$E = \frac{LE_0}{LE} + \frac{SI}{SI_0} \quad (23)$$

第2个优化目标为总能耗, 其计算公式为

$$TEC = \sum_{j=1}^{Nm} \sum_{k=1}^2 EC_{jk} \quad (24)$$

2 超启发式三维分布估计算法

在HH3DEDA中, 首先利用3DEDA对高层策略域中12种有效邻域操作的排列(其中每个排列为高层策略域内一个个体)进行优化, 进而获得高层

策略域种群并对其进行更新; 然后在低层问题域, 将每个高层个体作为一种新的启发式算法, 对低层问题域中相应个体进行邻域搜索。

2.1 编码与解码

2.1.1 高层策略域编码与解码

对于高层策略域, 种群中每个个体 P 均由12种LLHs排列构成, 个体长度为12, 且在同一个体中允许出现相同的低层启发式操作。解码高层策略域个体时, 对于低层问题域中的解, 从左到右依次执行高层策略域中的低层启发式操作, 当执行完所有低层启发式操作后, 高层策略域个体所得的适应值, 即为其对应低层问题域个体需更新的适应值。高层策略域个体示意图如图3所示。



图3 高层策略域个体示意图

Fig. 3 High-level policy domain individual diagram

2.1.2 低层问题域编码与解码

(1) 工序和机器人两段式编码

对于低层问题域, 每个个体即为原问题的1个解。低层问题域个体 π 的编码由工序序列和机器人序列两部分组成。但由于GRTALBP-I属于已知节拍时间而工作站数量未知的TALBP-I, 为使编码序列能够表明机器人的分配情况, 本文参照文献[21]的方法规定最大工位数 N 为其下限值的2倍, 即

$$N = 2 \left(\sum_{i \in I} t_i / CT \right) \quad (25)$$

工序编码为选择因子从大到小的排序, 长度为 $M_i = \sum i$ 。机器人序列的编码从编码序列的第 $M_i + 1$ 位开始, 长度为 N , 每2位代表相应成对工位上机器人的分配情况, 取值1~3分别表示不同的机器人类型, 工序和机器人的两段式编码如图4所示。

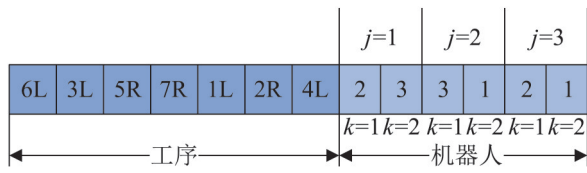


图 4 工序和机器人两段式编码
Fig. 4 Process and robot two-step coding

图 4 中，成对工位 1 的左右工位中分配的机器人类型分别为 2、3；成对工位 2 的左右工位中分配的机器人类型分别为 3、1；成对工位 3 的左右工位中分配的机器人类型分别为 2、1。

(2) 基于工序选择因子的组合编码

为提高编解码效率，本文针对 TALBP 的工序分配问题提出了一种基于工序选择因子的组合编码 (task selection factory based combination code, TSFBCC) 方式。TSFBCC 核心在于将工序分配的选择优先级信息与工序直接关联，从而减少计算复杂度；其中，组合编码是指工序分配向量 g 不再简单地代表该次序所要装配的任务号码，而是由两部分组成，分别是该次序所要装配的工序号码和方位。TSFBCC 原理如图 5 所示，具体描述如下：

步骤 1：算法初始化时为每个工序生成与其对应的 0~1 之间不重复的随机实数。

步骤 2：选择满足所有约束关系的候选工序中对应的随机实数最大的工序放入工位中，待所有

工序分配完成后，将每个工序的序号按照对应随机实数从大到小排序，得到工序分配子序列。

步骤 3：初始化选择因子，在运行算法前生成一组与工序规模相同的 0~1 之间不重复的随机实数，并将这组随机实数从大到小进行排序，该组实数就称为选择因子。

步骤 4：在算法优化过程中，选择因子将按照工序子序列对应匹配到每个工序，而对于所有满足约束条件的候选工序，将选择具有最大选择因子的工序放入工位中。

该编码中选择因子的值实际为工序分配优先关系的映射，算法对工序序列优化的本质即为获得更优工序分配优先关系的过程。运用此编码方式既减小了运算复杂度，又可体现双边装配线平衡问题对任务操作方位的要求，从而提高编解码效率。

(3) 解码

具体解码步骤如下：

步骤 1：初始化当前工序数为 $m = \sum_{i \in I} i$ ，可用机器人数量为 $num_R = NR$ ，工序集 $D = I$ ，按工序编码分配选择因子至相应工序。

步骤 2：开启一个新的成对工位 j ，更新可分配工序集 D_j 和机器人可分配集 Dr ，初始化可分配工序集 D_j 和 Dr 使其为空，对于所有未分配的工序 i ，执行 $D_j = D_j + i$ ；对于机器人 r ，执行 $Dr = Dr + r$ 。

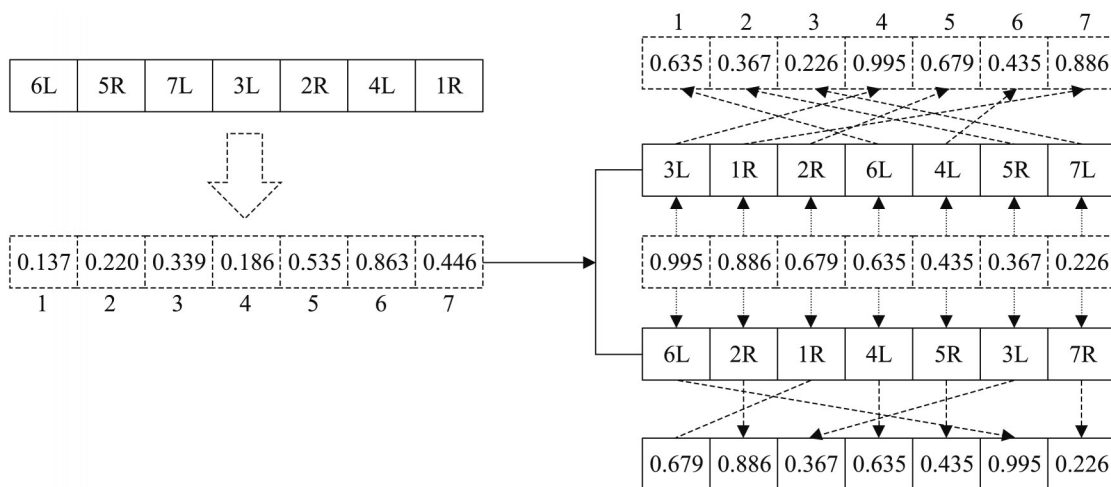


图 5 TSFBCC 编码原理图
Fig. 5 TSFBCC coding schematic

步骤3: 判断机器人是否可用, 若 $NR \leq num_R$, 则按照编码方式从 D_r 中选出2个机器人放入当前成对工位中, 若 $r_1=1$ 且 $k=1$, $r_2=3$ 且 $k=2$, 即表示成对工位 j 中, 左、右工位中分别分配了1, 3两种类型的机器人。

步骤4: 选择 D_j 中满足优先约束关系且具有最大选择因子的工序 i' , 判断工序 i' 是否在约束集内, 若 i' 在消极区域约束内, 转至步骤5; 若 i' 在积极区域约束内, 转至步骤6; 若 i' 在同步区域约束内, 转至步骤7; 否则, 转至步骤9。

步骤5: 判断工序 i' 是否满足步骤7, 若满足, 执行步骤9; 否则, 将该工序及其后序工序从可分配工序集 D_j 中删除, 并转至步骤10。

步骤6: 判断工序 i' 对应的其他各工序是否也在可分配工序集 D_j 中且满足步骤2、7, 若满足且符合节拍时间, 则根据这些工序的操作方位将其一并分配到同一工位内; 否则, 将这些工序及其后序工序从可分配工序集 D_j 中删除, 并转至步骤10。

步骤7: 判断工序 i' 对应的另一个工序 l 是否也在可分配工序集 D_j 中且满足步骤2, 若满足, 执行步骤8; 否则, 将这些工序及其后序工序从可分配工序集 D_j 中删除, 并转至步骤10。

步骤8: 判断工序 l 是否也在消极区域约束内, 若满足, 则判断其是否满足步骤7; 若不满足, 则将工序 i' , l 以及其后序工序从可分配工序集 D_j 中删除; 若满足或工序 l 不在消极区域约束内, 且符合节拍约束, 则根据其操作方位, 将其分配到两个相对的工位内, 执行步骤10, 否则, 将这些工序及其后序工序从可分配工序集 D_j 中删除, 并转至步骤10。

步骤9: 若工序 i' 满足步骤2且符合节拍时间, 则根据其操作方位, 将其分配到相应的工位上, 否则, 将其以及后序工序从可分配工序集 D_j 中删除, 并转至步骤10。

步骤10: 判断可分配工序集 D_j 是否为空, 若满足, 则转至步骤11; 否则, 转至步骤4。

步骤11: 判断工序序列中的工序是否已经全部分配完成, 即 D 是否已经清零, 若满足, 则转至步骤12; 否则, 转至步骤2。

步骤12: 判断是否可以在满足所有约束的情况下, 将最后一个成对工位上的工序全部分配到一个工位上, 若满足, 则调整到满足操作方位约束的工位上; 最后, 进行位置约束检查, 若符合则解码结束。

2.2 三维分布估计算法

EDA是一种基于统计学习理论的群智能优化算法, 已被广泛用于求解多种调度优化问题^[22-24]。然而, 现有EDA基本都采用二维概率模型来学习并积累优质解的信息。二维概率模型因其自身结构的制约, 无法保留优质解内由相邻工序所构成相似块的位置信息, 这导致采用该模型生成新个体时, 容易误导搜索方向。显然, 三维概率模型可利用多出的第三维下标保存位置信息, 有利于提高概率模型的引导能力。因此, 本文采用三维EDA作为HH3DEDA的高层算法。

2.2.1 块结构与相似块

在高层策略域中, 将每个高层个体序列中连续相邻的2个启发式操作视为一个块结构, 不同位置的块结构视为相似块。譬如, 以某一个高层个体 $\theta = \{8, 6, 5, 7, 6, 7, 8, 3, 4, 9, 10, 3\}$ 来说, $[8, 6]$ 即为一个块结构, $[8, 6]$ 、 $[6, 5]$ 、 $[5, 7]$ 等即为相似块。显然, 高层策略域中的个体由位于不同位置的块结构或一系列相互作用的相似块构成。因此, 本文结合相似块的分布特性, 设计了一种有效的概率模型用于引导高层策略域种群的更新。

2.2.2 三维矩阵模型

定义 $Pop(gen)$ 为3DEDA内第 gen 代高层策略域种群, ps 为 $Pop(gen)$ 的种群规模; $SPOP(gen)$ 为 $Pop(gen)$ 内的优质个体, sps 为 $SPOP(gen)$ 的种群规模, $SPOP(gen) = \{\Pi_{SPOP}^{gen,1}, \Pi_{SPOP}^{gen,2}, \dots, \Pi_{SPOP}^{gen,sps}\}$, $\Pi_{SPOP}^{gen,k}$ 为 $SPOP(gen)$ 内的第 k 个个体, $\Pi_{SPOP}^{gen,k} =$

$[\Pi_{SPop}^{gen,k}(1), \Pi_{SPop}^{gen,k}(2), \dots, \Pi_{SPop}^{gen,k}(n)]$ 。定义三维矩阵 $MC_{n \times n \times n}^{gen}$ 用于存储第 gen 代中优质个体内块结构及其分布信息，关于 $MC_{n \times n \times n}^{gen}(x, y, z)$ 的具体描述如下：

特征函数用于记录优质种群 $SPop(gen)$ 内个体 k 的启发式操作顺序和相似块的位置，计算公式为

$$N_{MC_{n \times n \times n}^{gen,k}}(x, y, z) = \begin{cases} 1, & y = \Pi_{SPop}^{gen,k}(x), \\ & \text{且 } z = \Pi_{SPop}^{gen,k}(x+1) \\ 0, & \text{否则} \end{cases}$$

$$x = 1, 2, \dots, n-1, y = 1, 2, \dots, n, z = 1, 2, \dots, n, k = 1, 2, \dots, sps \quad (26)$$

记录函数用于记录高层策略域中所有优质个体的启发式操作顺序、块结构的位置信息和 $SPop(gen)$ 内块结构 $[y, z]$ 在 x 位置上出现的次数，计算公式为

$$MC_{n \times n \times n}^{gen}(x, y, z) = \sum_{k=1}^{sps} N_{MC_{n \times n \times n}^{gen,k}}(x, y, z)$$

$$x = 1, 2, \dots, n-1, y = 1, 2, \dots, n, z = 1, 2, \dots, n \quad (27)$$

三维矩阵 $MC_{n \times n \times n}^{gen}(x, y, z)$ 的内部结构描述为

$$MC_{n \times n \times n}^{gen}(x) = \begin{bmatrix} MC_{n \times n \times n}^{gen}(x, 1) \\ \vdots \\ MC_{n \times n \times n}^{gen}(x, n) \end{bmatrix}_{n \times 1} = \begin{bmatrix} MC_{n \times n \times n}^{gen}(x, 1, 1) & \dots & MC_{n \times n \times n}^{gen}(x, 1, n) \\ \vdots & & \vdots \\ MC_{n \times n \times n}^{gen}(x, n, 1) & \dots & MC_{n \times n \times n}^{gen}(x, n, n) \end{bmatrix}_{n \times n} \quad (28)$$

$$MC_{n \times n \times n}^{gen}(y, z) = [MC_{n \times n \times n}^{gen}(1, y, z), MC_{n \times n \times n}^{gen}(2, y, z), \dots, MC_{n \times n \times n}^{gen}(n, y, z)]_{1 \times n}$$

$$x = 1, 2, \dots, n-1, y = 1, 2, \dots, n, z = 1, 2, \dots, n \quad (29)$$

2.2.3 三维概率模型

本文在二维概率模型的基础上提出了一种基于三维概率模型的进化策略，并设计了一种基于增量学习的更新机制来有效地累积优质高层排列信息。在 HH3DEDA 中，使用基于 $MC_{n \times n \times n}^{gen}$ 的三维概率模型 $P_{MC_{n \times n \times n}^{gen}}$ ，用于学习并积累 $SPop(gen)$ 内相似块的位置信息，其中 $P_{MC_{n \times n \times n}^{gen}}(x, y, z)$ 代表相似块 $[y, z]$ 在 x 位置上分布的概率，具体描述为

$$P_{MC_{n \times n \times n}^{gen}}(x) = \begin{bmatrix} P_{MC_{n \times n \times n}^{gen}}(x, 1) \\ \vdots \\ P_{MC_{n \times n \times n}^{gen}}(x, n) \end{bmatrix}_{n \times 1} = \begin{bmatrix} P_{MC_{n \times n \times n}^{gen}}(x, 1, 1) & \dots & P_{MC_{n \times n \times n}^{gen}}(x, 1, n) \\ \vdots & & \vdots \\ P_{MC_{n \times n \times n}^{gen}}(x, n, 1) & \dots & P_{MC_{n \times n \times n}^{gen}}(x, n, n) \end{bmatrix}_{n \times n} \quad (30)$$

定义 $Sum_P_{MC_{n \times n \times n}^{gen}}(x)$ 为高层策略域内所有优质个体中相似块 $[y, z]$ 在 x 位置分布的概率和：

$$Sum_P_{MC_{n \times n \times n}^{gen}}(x) = \sum_{y=1}^n \sum_{z=1}^n P_{MC_{n \times n \times n}^{gen}}(x, y, z) \quad (31)$$

定义 $Sum_MC_{n \times n \times n}^{gen}(x)$ 为高层策略域内所有优质个体中相似块 $[y, z]$ 在 x 位置上分布的次数之和：

$$Sum_MC_{n \times n \times n}^{gen}(x) = \sum_{y=1}^n \sum_{z=1}^n MC_{n \times n \times n}^{gen}(x, y, z) \quad (32)$$

关于概率模型 $P_{MC_{n \times n \times n}^{gen}}$ 的具体更新过程如下：

步骤 1：首先对概率模型 $P_{MC_{n \times n \times n}^0}$ 进行初始化：

$$P_{MC_{n \times n \times n}^0}(x, y, z) = \begin{cases} 1/n, & x = 1, y, z = 1, 2, \dots, n \\ 1/n^2, & x = 2, 3, \dots, n-1, y, z = 1, 2, \dots, n \end{cases} \quad (33)$$

步骤 2：计算 $MC_{n \times n \times n}^0$ 并更新概率模型：

$$P_{MC_{n \times n \times n}^1}(x, y, z) = \begin{cases} \frac{MC_{n \times n \times n}^0(x, y, z)}{Sum_MC^0(x)}, & x = 1, y, z = 1, 2, \dots, n \\ (P_{MC_{n \times n \times n}^1}(x, y, z) + \frac{MC_{n \times n \times n}^0(x, y, z)}{Sum_P_{MC}^0(x) + Sum_MC^0(x)}) & x = 2, 3, \dots, n-1, y, z = 1, 2, \dots, n \end{cases} \quad (34)$$

步骤 3：定义 $gen = 1$ 。

步骤 4：计算 $MC_{n \times n \times n}^{gen}$ 并更新概率模型。

$$P_{MC_{n \times n \times n}^{gen+1}}(x, y, z) = (1-r)P_{MC_{n \times n \times n}^{gen}}(x, y, z) + r \cdot MC_{n \times n \times n}^{gen}(x, y, z) / Sum_MC^{gen}(x),$$

$$x = 1, 2, \dots, n-1, y, z = 1, 2, \dots, n \quad (35)$$

步骤 5：令 $gen = gen + 1$ ，若 $gen < Max\ gen$ ，则转至步骤 4。

2.3 种群初始化

对于高层策略域, 按照式(30)对初始三维概率模型进行采样生成高层初始种群。为获得较高质量的初始解, 本文设计了一种改进的NEH启发式算法来生成低层问题域的初始种群, 具体描述如下:

步骤1: 基于快速更新最优分配(IUFF)启发式算法^[25]来生成低层问题域个体 π 的工序排序, $\pi = \{\pi(1), \pi(2), \dots, \pi(n)\}$ 。

步骤2: 序列 π 中的 $\pi(1)$ 与 $\pi(2)$ 进行位置互换, 并生成新的序列 π^1 , 并将两个解进行比较, 若新解可支配旧解, 则用新解替换旧解; 否则, 保留旧解。

步骤3: 将工序 $\pi(i)$ ($i=3, 4, \dots, Nt$)依次插入至原始位置 i 之前的所有可能位置, 并取其中最优化位置来更新个体 π 。

2.4 高层策略域种群采样更新机制

在高层策略域中, 定义 $\Pi^{gen,k} = [\Pi^{gen,k}(1), \Pi^{gen,k}(2), \dots, \Pi^{gen,k}(n)]$ 为 $Pop(gen)$ 内的第 k 个个体, $Select(\Pi^{gen,k}, i)$ (其中, $i > 1$)为采样选择函数, 用于选择在 $\Pi^{gen,k}$ 中 i 位置所出现的低层启发式操作。相似块 $[\Pi^{gen,k}(i-1), \Pi^{gen,k}(i)]$ 被选中的概率存储在三维概率模型 $P_MC_{n \times n \times n}^{gen-1}(i-1)$ 内, 因此 $Select(\Pi^{gen,k}, i)$ 依据三维概率模型 $P_MC_{n \times n \times n}^{gen-1}(i-1)$ 进行采样, $Select(\Pi^{gen,k}, i)$ 的具体操作过程如下:

步骤1: 生成随机概率数 r

$$r \in \left[0, \sum_{h=1}^n P_MC_{n \times n \times n}^{gen-1}(i-1, \Pi^{gen,k}(i-1), h) \right];$$

步骤2: 用轮盘赌的方法选择 $LLHs$

$$\text{若 } r \in [0, P_MC_{n \times n \times n}^{gen-1}(i-1, \Pi^{gen,k}(i-1), 1)],$$

$LLHs = 1$, 需转至步骤3;

$$\text{若 } r \in \left[\sum_{h=1}^{pos} P_MC_{n \times n \times n}^{gen-1}(i-1, \Pi^{gen,k}(i-1), h), \right.$$

$$\left. \sum_{h=1}^{pos+1} P_MC_{n \times n \times n}^{gen-1}(i-1, \Pi^{gen,k}(i-1), h) \right],$$

$pos \in \{1, 2, \dots, n-1\}$, 则 $LLHs = pos + 1$;

步骤3: 返回 $LLHs$

由于高层策略域内个体在初始位置1处不存在相似块 $[\Pi^{gen,k}(i-1), \Pi^{gen,k}(i)]$, 导致 $Select(\Pi^{gen,k}, i)$ 无法在位置1处进行采样。因此本文对初始位置1设计了一种新的采样策略, 具体描述如下:

步骤1: 依据式(36)计算 $Sum_P_MC_{n \times n \times n}^{gen-1}(y)$ 。

$$Sum_P_MC_{n \times n \times n}^{gen-1}(y) = \sum_{z=1}^n P_MC_{n \times n \times n}^{gen-1}(1, y, z) \quad (36)$$

$$y = 1, 2, \dots, n$$

步骤2: 采用轮盘赌的方法来选择 $\Pi^{gen,k}$ ($k=1, 2, \dots, ps$)中初始位置1的低层启发式操作, 随机生成概率数 r' 且 $r' \in \left[0, \sum_{h=1}^{pos} Sum_P_MC_{n \times n \times n}^{gen-1}(1) \right]$ 。若 $r' \in [0, Sum_P_MC_{n \times n \times n}^{gen-1}(1)]$, 则 $s=1$, 转至步骤3; 若 $r' \in \left[\sum_{h=1}^{pos} Sum_P_MC_{n \times n \times n}^{gen-1}(h), \sum_{h=1}^{pos+1} Sum_P_MC_{n \times n \times n}^{gen-1}(h) \right]$, $pos \in \{1, 2, \dots, n-1\}$, 则 $s = pos + 1$ 。

步骤3: 令 $\Pi^{gen,k}(1)$ ($k=1, 2, \dots, ps$)为 $LLHs$ 。

基于上述描述, 高层策略域种群 $Pop(gen)$ 的采样更新过程如下:

步骤1: 定义 $k=1$ 。

步骤2: 基于初始位置 $i=1$ 处的采样策略选择 $\Pi^{gen,k}(1)$ 的低层启发式操作 $LLHs$ 。

步骤3: 令 $i=2$ 。

步骤4: 令 $\Pi^{gen,k}(i) = Select(\Pi^{gen,k}, i)$ 。

步骤5: 若 $i \leq n$, 则令 $i=i+1$, 并转至步骤4。

步骤6: 若 $k \leq ps$, 则令 $k=k+1$, 并转至步骤2。

步骤7: 输出高层策略域种群 $Pop(gen)$ 。

2.5 低层启发式操作

为增强HH3DEDA对解空间内优质非支配解的搜索能力, 本文在算法低层设计了多种有效邻域操作, 并通过动态混合构成多种不同的启发式

算法,可在算法搜索到达多种邻域结构共同的局部最优非支配解前保持一直向下搜索的状态,从而增强算法对解空间的搜索深度,有利于算法获得优质非支配解集。具体来说,本文基于求解组合优化问题常用的有效邻域操作(交换,插入等),设计了如下12种邻域操作,作为低层启发式操作。

(1) LLH1: 工序作业序列交叉操作。从工序作业序列内随机选择两位工序进行位置交换。

(2) LLH2: 工序作业序列前项插入操作。从工序作业序列内随机选择两位工序,将位置靠后的工序插入到位置靠前的工序之前。

(3) LLH3: 工序作业序列相邻工序交叉操作。从工序作业序列内随机选择一位工序,将其与向前一位或向后一位的工序进行交换,且向前或向后交换的概率相等。

(4) LLH4: 工序作业序列变异操作。从工序作业序列内随机选择一位工序,将其变异为序列内任意一位其他工序。

(5) LLH5: 工序作业序列逆序操作。从工序作业序列内随机选择两位工序,将两位之间以及所选的两位工序的序列进行逆序排列。

(6) LLH6: 学习优质作业序列操作。从工序作业序列内随机截取一段,并将其替换为优质个体中相同位置的作业序列。

(7) LLH7: 机器人序列交叉操作。从机器人序列内随机选择两位机器人进行位置交换。

(8) LLH8: 机器人序列前项插入操作。从机器人序列内随机选择两位机器人,将位置靠后的机器人插入到位置靠前的机器人之前。

(9) LLH9: 机器人序列相邻工序交叉操作。从机器人序列内随机选择一位机器人,将其与向前一位或向后一位的机器人进行交换,且向前或向后交换的概率相等。

(10) LLH10: 机器人序列贪婪操作。从机器人序列内随机选择两位机器人,依次对其进行插空操作。

(11) LLH11: 机器人序列逆序操作。从机器人序列内随机选择两位机器人,将两位之间以及所选两位机器人的序列进行逆序操作。

(12) LLH12: 学习优质机器人序列操作。从机器人序列内随机截取一段,并将其替换为优质个体中相同位置的机器人序列。

2.6 机器人关机节能策略

对于每个工位上的机器人而言,当机器人 r 处于待机状态且此时由待机所产生能耗大于所设定的阈值 $EC_{\text{off-on}}$ 时,则考虑执行关机策略来降低能耗,但执行关机策略时必须不影响下个工序的开始作业时间。

具体描述如下:

步骤1: 判断工位 (j, k) 中机器人 r 当前由待机所产生的能耗是否大于关机所需能耗 $EC_{\text{off-on}}$,若满足,则转至步骤2;否则,不执行节能策略。

步骤2: 判断工位 (j, k) 中机器人 r 的当前待机时间是否大于关机所需时间 $T_{\text{min},r}$,若满足,则执行机器人关机策略;否则,不执行节能策略。

2.7 算法流程

根据以上描述,HH3HEDA算法的流程如图6所示。

具体流程如下:

步骤1: 按照改进的NEH规则初始化低层问题域种群,种群大小为 ps ;初始化三维概率模型。

步骤2: 评价低层策略域种群。

步骤3: 采样生成高层策略域种群。

步骤4: 将高层策略域个体中每一位所对应的启发式操作来依次对低层问题域个体进行更新,每执行完一次启发式操作后,就对新生成的新解与旧解进行比较,若新解能够支配旧解,将旧解替换为新解;否则,则以一定的概率来接受新解,并继续执行剩余的启发式操作。执行完高层策略域个体中所有的启发式操作后,此高层个体的适应值即为对应低层个体的适应值。

步骤 5: 判断高层个体的支配等级并计算该个体的拥挤距离; 对其进行排序后选择当前种群中前 $ps \times \rho$ 个个体来更新三维概率模型。

步骤 6: 判断是否满足终止条件, 若不满足则转至步骤 2; 否则, 终止循环。

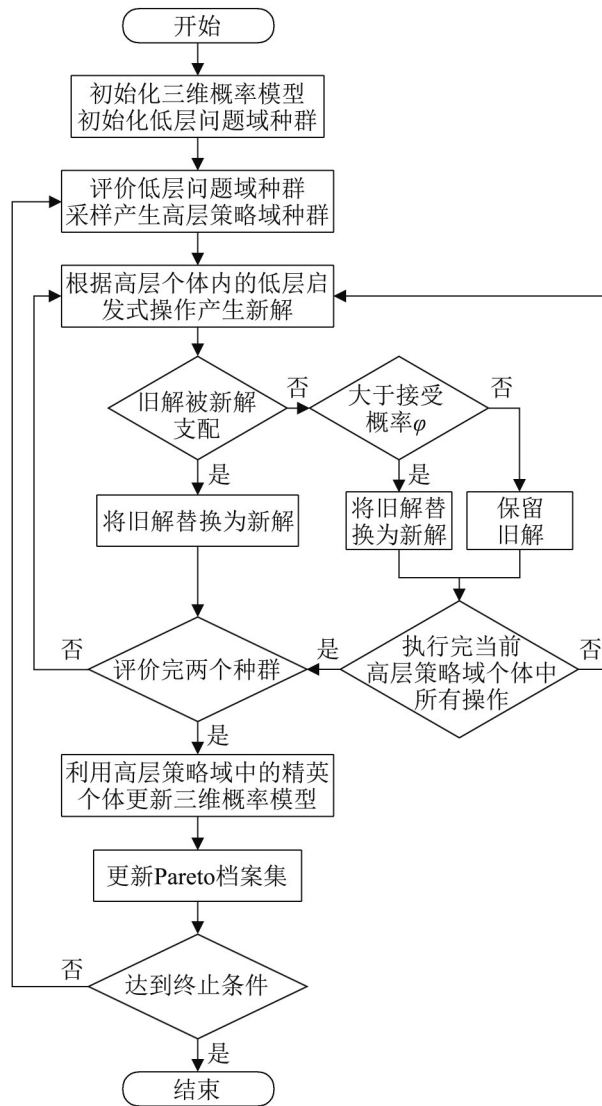


图 6 HH3DEDA 流程图
Fig. 6 HH3DEDA flow chart

3 实验设计与分析

3.1 相关数据设置

本文采用若干规模的双边装配线平衡标准算例^[8]来对 GRTALBP-I 进行相关实验和分析, 其中

包含小规模算例 3 组: P12、P16 和 P24; 大规模算例 3 组: P65、P148 和 P205。问题中相关资源数据的设置, 参考文献[26, 14]提出的方法。机器人对工序的作业时间按式(37)进行计算:

$$t_{ir} = (100 - \varepsilon)t_{i0}, \quad \forall r \in R \quad (37)$$

式中: ε 为机器人对工序作业时间的改善效率, 且不同类型机器人对不同工序作业时间的改善效率 ε 不同。

表 2 给出了不同类型机器人对应的 ε 和能耗区间; 表 3 是不同算例规模下机器人的数量约束; 表 4 则表示标准算例 P65 的部分数据设置; ε 在区间内随机生成。

表 2 不同类型机器人对应的效率和能耗
Table 2 Corresponding efficiency and energy consumption of different types of robots

机器人类型	效率 ε	能耗
R_1	$\varepsilon \sim U(10, 30)$	$EC_{R_1} \in (0.5, 2.0)$
R_2	$\varepsilon \sim U(20, 40)$	$EC_{R_2} \in (1.0, 2.5)$
R_3	$\varepsilon \sim U(30, 50)$	$EC_{R_3} \in (1.5, 3.0)$

表 3 不同算例规模下机器人数量约束
Table 3 Constraints on the number of robots in different example sizes

I	NR_1	NR_2	NR_3	NR
12	4	3	3	10
16	4	3	3	10
24	6	4	4	14
65	7	5	6	18
148	8	7	7	22
205	10	7	7	24

表 4 标准算例 P65 的部分数据设置
Table 4 Part data Settings of standard example P65

i	优先约束工序	t_{i0}	ε_1	ε_2	ε_3
1	—	49	13	25	36
3	1, 2	71	18	27	35
16	15	151	21	32	39
25	4	113	26	28	41
35	34	132	17	36	43
46	44, 45	37	23	35	36
55	31	47	25	21	37
60	31	25	27	36	45

3.2 参数设置

GRTALBP-I 中参数取值如下：机器人 r 单位时间内的空载能耗为 $SEC_r=0.1EC_r$ ，机器人执行开关机所需能耗为 $EC_{off-on}=2$ ，执行一次开关机的时间为 $T_{min_r}=10$ ， LE_0 设为 100， SI_0 对应于算例 P12, P16, P24 和 P65 设为 25，P148 和 P205 分别设为 45 和 300。种群规模 ps 设置为 40，统计样本的分位数 ρ 设置为 0.1，学习率 r 设置为 0.6。

3.3 仿真结果与比较

为更客观地描述不同算法所产生的非支配解集 S_j ，本文采用文献[27]提出的方法对不同算法产生的非支配解集进行评价，描述如下：

$$R_N(S_j) = \frac{|S_j - \{x \in S_j | \exists y \in S: y \prec x\}|}{|S_j|} \quad (38)$$

$$N_N(S_j) = |S_j - \{x \in S_j | \exists y \in S: y \prec x\}| \quad (39)$$

式中：指标 R_N 为 S_j 中不被 S 中任意解所支配的解所占比例，用于衡量算法所得非支配解的质量；指标 N_N 为解集 S_j 不被解集 S 中任意解所支配的数量，用于衡量算法获得非支配解的能力。显然， $R_N(S_j)$ 和 $N_N(S_j)$ 的值越大，算法性能越好。

本文所有算法和实验均由 Delphi2010 编程实现，操作系统为 Windows10，内存为 16 GB，CPU 为 i7-8565U，主频为 1.99 GHz。对于不同规模的算例，设定每种算法的运行时间为 $2n$ s， n 为算例规模，所有测试算例均独立运行 20 次，每个算例对应的最优结果用粗体表示。

3.3.1 高层策略的有效性验证

为验证 HH3DEDA 中 3DEAD 作为高层策略的有效性，将 HH3DEDA 与采用 EDA 作为高层策略的超启发式分布估计算法 (hyper-heuristic estimation of distribution algorithm, HHEDA) 以及采用遗传算法 (genetic algorithm, GA) 作为高层策略的超启发式遗传算法 (hyper-heuristic genetic algorithm, HHGA) 进行比较。在比较过程中，这三

种算法除采用的高层策略不同外，其余部分均相同，比较结果如表 5 所示。

表 5 验证 HH3DEDA 高层策略的有效性
Table 5 Effectiveness of verifying HH3DEDA high-level policies

n	CT/s	HHGA		HHEDA		HH3DEDA	
		R_N	N_N	R_N	N_N	R_N	N_N
12	5	0.568 6	0.75	0.650 2	1.10	0.871 7	5.75
12	6	0.495 0	1.20	0.607 4	2.90	0.811 7	6.30
12	8	0.595 8	0.80	0.687 6	0.95	0.825 9	6.50
12	9	0.550 0	1.70	0.614 7	2.20	0.897 1	6.85
16	15	0.538 3	2.20	0.636 7	2.75	0.863 3	5.70
16	16	0.421 0	1.50	0.383 6	1.50	0.768 3	5.95
16	20	0.391 7	1.85	0.408 3	3.50	0.777 4	4.60
16	21	0.409 3	1.70	0.546 7	2.70	0.826 7	4.80
24	20	0.370 5	1.50	0.688 7	1.70	0.847 3	5.35
24	25	0.421 0	2.00	0.623 4	3.00	0.783 3	6.10
24	30	0.316 7	1.15	0.587 0	2.80	0.821 9	5.85
24	40	0.393 3	1.00	0.665 0	2.35	0.885 2	5.20
65	326	0.523 3	2.10	0.790 5	2.40	0.585 0	3.50
65	381	0.501 7	2.20	0.525 8	3.10	0.704 9	4.65
65	435	0.346 9	1.40	0.753 3	2.70	0.877 4	3.80
65	490	0.601 1	2.25	0.637 1	3.25	0.830 2	5.45
65	544	0.461 9	1.90	0.734 5	3.20	0.682 5	3.45
148	255	0.466 7	1.30	0.678 3	2.80	0.813 0	4.10
148	306	0.568 3	2.35	0.580 0	2.10	0.760 5	3.95
148	357	0.364 3	1.10	0.734 8	2.90	0.801 7	4.05
148	459	0.475 0	1.80	0.768 3	3.10	0.641 2	2.90
148	510	0.686 6	2.45	0.641 4	3.00	0.826 6	4.50
205	188 8	0.617 8	2.50	0.615 0	2.40	0.733 3	3.30
205	226 6	0.541 7	1.50	0.672 6	1.90	0.600 0	3.15
205	245 4	0.586 7	1.40	0.480 0	1.30	0.758 3	2.25
205	264 3	0.436 7	1.50	0.425 0	1.70	0.747 6	3.50
205	283 2	0.347 6	1.00	0.503 3	1.60	0.825 0	2.70

由表 5 可知，HH3DEDA 在绝大部分算例上的测试结果都明显优于 HHEDA 和 HHGA，由此验证了 3DEDA 作为高层策略的有效性。实际上，3DEDA 和 EDA 这类基于概率模型的算法，可在一定程度上避免 GA 这类传统进化算法中普遍存在的对较优解模式破坏的问题^[18]。3DEDA 相对于常规 EDA 而言，可以有效引导由相邻 2 个启发式操作所构成的块结构的放置位置，因此可以更好

地保留优质解的信息并合理引导搜索方向, 故 HH3DEDA 在上述实验中表现优异。

3.3.2 本文算法的有效性验证

为验证 HH3DEDA 算法的有效性, 本节将 HH3DEDA 与近年来重要国际期刊中的多目标优化算法进行对比, 包括 ITLBO^[8], IMMOGLS^[27]。

ITLBO 是一种求解多目标 TALBP-I 的有效算法。IMMOGLS 是一种基于随机权重的多目标遗传

算算法。在此节实验中, 由于对比算法中均带有局部搜索操作, 故将算法的单次运行时间增加至 $3ns$, n 为算例规模。由表 6 可知, HH3DEDA 对绝大部分问题的求解性能均优于其他对比算法, 验证了算法在求解 GRTALBP-I 的有效性。图 7、8 为各算法求解部分算例所得非支配解示意图, 显然 HH3DEDA 所获取的非支配解的质量、分散性与多样性较优。

表 6 算法比较结果
Table 6 Algorithm comparison results

n	CT/s	ITLBO		IMMOGLS		HH3DEDA		n	CT/s	ITLBO		IMMOGLS		HH3DEDA	
		R_N	N_N	R_N	N_N	R_N	N_N			R_N	N_N	R_N	N_N	R_N	N_N
12	5	0.223 6	0.85	0.041 7	0.25	0.9272	6.25	65	435	0.140 7	0.70	0.070 4	0.25	0.960 8	5.75
12	6	0.020 0	0.15	0.156 7	0.75	0.966 7	7.65	65	490	0.116 3	0.40	0.086 7	0.25	0.933 7	5.25
12	8	0	0	0	0	1.000 0	6.65	65	544	0.151 7	0.45	0.095 1	0.40	0.849 1	6.05
12	9	0.083 3	0.35	0	0	0.971 4	6.30	148	255	0.154 2	0.55	0.112 8	0.60	0.916 3	7.45
16	15	0.223 8	0.90	0.044 6	0.25	0.962 8	7.45	148	306	0.170 4	0.45	0.092 5	0.40	0.965 6	6.55
16	16	0.098 2	0.55	0.125 6	0.45	0.887 1	6.30	148	357	0.186 7	0.60	0.223 2	1.00	0.891 7	6.05
16	20	0.075 0	0.10	0.035 0	0.10	0.947 5	6.35	148	459	0.191 8	0.90	0.111 7	0.40	0.940 8	7.95
16	21	0.202 1	0.50	0.666 7	0.20	0.962 6	8.40	148	510	0.148 0	0.60	0.114 3	0.45	0.925 1	7.15
24	20	0.154 6	0.65	0.215 4	0.60	0.883 3	6.40	205	188 8	0.033 8	0.20	0.477 1	2.00	0.849 2	3.85
24	25	0.104 5	0.50	0.162 1	0.60	0.907 6	6.10	205	226 6	0.006 3	0.05	0.326 7	0.65	0.858 3	3.45
24	30	0.076 7	0.30	0.023 8	0.10	0.990 0	5.70	205	245 4	0.112 5	0.15	0.277 5	1.05	0.915 0	3.55
24	40	0.018 3	0.10	0	0	1.000 0	5.35	205	264 3	0.050 0	0.10	0.104 2	0.25	0.969 1	4.20
65	326	0.161 7	0.65	0.128 0	0.40	0.971 7	6.10	205	283 2	0.050 0	0.05	0.079 2	0.25	1.000 0	4.85
65	381	0.070 8	0.30	0.071 2	0.35	0.973 3	6.85								

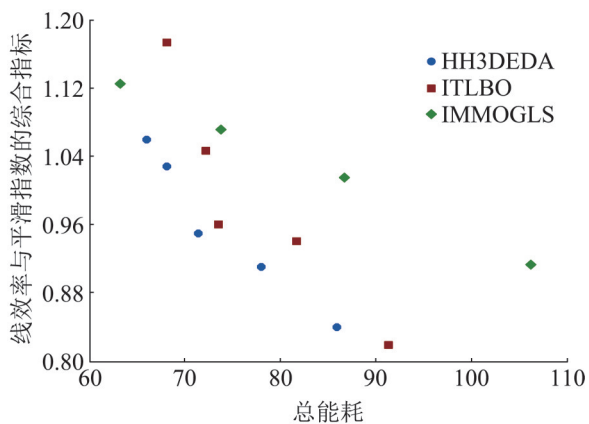


图 7 各算法求解 $n=16, CT=16$ 算例所得非支配解
Fig. 7 Each algorithm solves $n=16, CT=16$ and obtains the schematic diagram of non-dominated solution

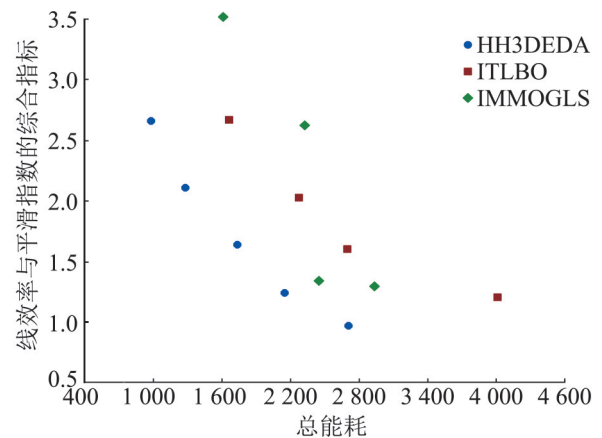


图 8 各算法求解 $n=148, CT=357$ 算例所得非支配解
Fig. 8 Each algorithm solves $n=148, CT=357$ and obtains the schematic diagram of non-dominated solution

综上，以上几种常规群智能算法每代均是采用数个固定的交叉、变异等邻域操作的方式来生成新种群以实现搜索，其搜索能力较为有限，而 HH3DEDA 利用高层的三维概率模型动态控制低层每个个体执行包含多个邻域操作的启发式搜索，有利于引导算法在多个不同区域进行较深入搜索，较容易发现复杂解空间中的优质解。因此，HH3DEDA 可在上述测试问题中取得较好结果。

4 结论

本文针对双边装配线平衡问题(TALBP)，考虑实际生产过程中机器人的作业效率与能耗问题，以同时最小化线效率及平滑指数的综合指标与总能耗为优化目标，提出了一种超启发式三维分布估计算法进行求解。

通过不同规模的仿真实验与算法比较，验证了 HH3DEDA 算法可有效求解 GRTALBP-I 问题。未来研究将把 HH3DEDA 扩展用于求解不确定多资源约束双边装配线平衡问题。

参考文献:

- [1] Salvesson M E. The Assembly Line Balancing Problem [J]. *Journal of Industrial Engineering* (S2314-4882), 1955, 29(10): 55-101.
- [2] Li Zixiang, Ibrahim Kucukkoc, J Mukund Nilakantan. Comprehensive Review and Evaluation of Heuristics and Meta-Heuristics for Two-Sided Assembly Line Balancing Problem[J]. *Computers and Operations Research* (S0305-0548), 2017, 84: 146-161.
- [3] Muhammad Razif Abdullah Make, Mohd Fadzil Faisae Ab Rashid, Muhamad Magffierah Razali. A Review of Two-Sided Assembly Line Balancing Problem[J]. *The International Journal of Advanced Manufacturing Technology* (S0268-3768), 2017, 89: 1743-1763.
- [4] Li Zixiang, Tang Qihua, Zhang LiPing. Two-Sided Assembly Line Balancing Problem of Type I: Improvements, a Simple Algorithm and a Comprehensive Study[J]. *Computers & Operations Research* (S0305-0548), 2017, 79: 78-93.
- [5] Bartholdi J J. Balancing Two-Sided Assembly Lines: A Case Study[J]. *International Journal of Production Research* (S0020-7543), 1993, 31(10): 2447-2461.
- [6] Zhong Yuguang, Deng Zexiao, Xu Ke. An Effective Artificial Fish Swarm Optimization Algorithm for Two-Sided Assembly Line Balancing Problems[J]. *Computers & Industrial Engineering* (S0360-8352), 2019, 138: 106-121.
- [7] Zheng Xiaojun, Ning Shiduo, Sun Hao, et al. Solving Multi-objective Two-Sided Assembly Line Balancing Problems by Harmony Search Algorithm Based on Pareto Entropy[J]. *IEEE Access* (S2169-3536), 2021, 9: 121728-121742.
- [8] Li Dashuang, Zhang Chaoyong, Shao Xinyu, et al. A Multi-objective TLBO Algorithm for Balancing Two-Sided Assembly line with Multiple Constraints[J]. *Journal of Intelligent Manufacturing* (S0956-5515), 2016, 27(4): 725-739.
- [9] 吴尔飞, 金焯, 汪峥. 双边装配线第二类平衡问题研究 [J]. *计算机集成制造系统*, 2005, 4(11): 1604-1608.
Wu Erfei, Jin Ye, Wang Zheng. Study on the Second Kind of Balance Problem of Bilateral Assembly Line[J]. *Computer Integrated Manufacturing Systems*, 2005, 4 (11): 1604-1608.
- [10] Kucukkoc Ibrahim, Zhang David Z. Type-E Parallel Two-Sided Assembly Line Balancing Problem: Mathematical Model and Ant Colony Optimisation Based Approach with Optimized Parameters[J]. *Computers and Industrial Engineering* (S0360-8352), 2015, 84: 56-69.
- [11] Li Zixiang, Nilanjan Dey, Amira S Ashour, et al. Discrete Cuckoo Search Algorithms for Two-Sided Robotic Assembly Line Balancing Problem[J]. *Neural Comput & Applic* (S0941-0643), 2018, 30: 2685-2696.
- [12] Li Zixiang, Mukund Nilakantan Janardhanan, Tang Qihua, et al. Model and Metaheuristics for Robotic Two-Sided Assembly Line Balancing Problems with Setup Times-ScienceDirect[J]. *Swarm and Evolutionary Computation* (S2210-6502), 2019, 50: 567-583.
- [13] Wang Kaipu, Li Xinyu, Gao Liang. A Multi-objective Discrete Flower Pollination Algorithm for Stochastic Two-Sided Partial Disassembly Line Balancing Problem [J]. *Computers & Industrial Engineering* (S0360-8352), 2019, 130: 634-649.
- [14] Li Zixiang, Tang Qihua, Zhang Liping. Minimizing Energy Consumption and Cycle Time in Two-Sided Robotic Assembly Line Systems Using Restarted Simulated Annealing Algorithm[J]. *Journal of Cleaner Production* (S0959-6526), 2016, 135: 508-522.
- [15] Lin Jian. Backtracking Search Based Hyper-Heuristic for the Flexible Job-Shop Scheduling Problem with Fuzzy Processing Time[J]. *Engineering Applications of Artificial Intelligence* (S0952-1976), 2019, 77(1):

- 186-196.
- [16] Song Hongbo, Lin Jian. A Genetic Programming Hyper-Heuristic for the Distributed Assembly Permutation Flow-Shop Scheduling Problem with Sequence Dependent Setup Times[J]. *Swarm and Evolutionary Computation* (S2210-6502), 2021, 60: 100807.
- [17] 钱斌, 余明哲, 胡蓉, 等. 超启发式交叉熵算法求解模糊分布式流水线绿色调度问题[J]. *控制与决策*, 2021, 36(6): 1387-1396.
- Qian Bin, She Mingzhe, Hu Rong, et al. Hyper-Heuristic Cross-Entropy Algorithm for Green Distributed Permutation Flow-Shop Scheduling Problem with Fuzzy Processing Time[J]. *Journal of Control and Decision*, 2021, 36(6): 1387-1396.
- [18] 杨海军, 李建武, 李敏强. 进化算法的模式、涌现与困难性研究[M]. 北京: 科学出版社, 2012.
- Yang Haijun, Li Jianwu, Li Minqiang. *Evolutionary Algorithms: Schema, Emergence and Hardness*[M]. Beijing: Science Press, 2012.
- [19] 李大双, 张超勇, 邵新宇, 等. 基于殖民竞争算法的多约束双边装配线平衡[J]. *机械工程学报*, 2015, 51(2): 183-189.
- Li Dashuang, Zhang Chaoyong, Shao Xinyu, et al. Hybrid Colonial Competitive Algorithm for the Two-Sided Assembly Line Balancing Problem with Multiple Constraints[J]. *Journal of Mechanical Engineering*, 2015, 51(2): 183-189.
- [20] 李大双, 张超勇, 邵新宇, 等. 基于多目标殖民竞争算法的随机型双边装配线[J]. *计算机集成制造系统*, 2014, 20(11): 2774-2787.
- Li Dashuang, Zhang Chaoyong, Shao Xinyu, et al. Balancing Stochastic Two-Sided Assembly Line with Multi-objective Colonial Competitive Algorithm[J]. *Computer Integrated Manufacturing Systems*, 2014, 20(11): 2774-2787.
- [21] 张子凯, 唐秋华, 张利平. 多重资源约束下的多目标U型装配线平衡[J]. *计算机集成制造系统*, 2017, 23(11): 2431-2441.
- Zhang Zikai, Tang Qiuhua, Zhang Liping. Multi-objective U-Shaped Assembly Line Balancing under Multi-resource Restriction[J]. *Computer Integrated Manufacturing Systems*, 2017, 23(11): 100-110.
- [22] 杨晓林, 胡蓉, 钱斌, 等. 增强分布估计算法求解低碳分布式流水线调度[J]. *控制理论与应用*, 2019, 36(5): 803-815.
- Yang Xiaolin, Hu Rong, Qian Bin, et al. Enhanced Distribution Estimation Algorithm for Low Carbon Distributed Pipeline Scheduling[J]. *Control Theory & Applications*, 2019, 36(5): 803-815.
- [23] Pan Quanke, Rubén Ruiz. An Estimation of Distribution Algorithm for Lot-Streaming Flow Shop Problems with Setup Times[J]. *Omega: The International Journal of Management Science* (S0305-0483), 2011, 40(2): 166-180.
- [24] Du Yu, Li Junqing, Luo Chao, et al. A Hybrid Estimation of Distribution Algorithm for Distributed Flexible Job Shop Scheduling with Crane Transportations[J]. *Swarm and Evolutionary Computation* (S2210-6502), 2021, 62: 861-885.
- [25] Hackman S T, Magazine M J, Wee T S. Fast, Effective Algorithms for Simple Assembly Line Balancing Problems[J]. *Operation Research* (S0030-364X), 1989, 37(6): 916-924.
- [26] 郑逸凡, 钱斌, 胡蓉, 等. CE-GA协同进化算法求解人机共同作业的U形装配线平衡问题[J]. *机械工程学报*, 2020, 56(9): 199-214.
- Zheng Yifan, Qian Bin, Hu Rong, et al. CE-GA Co-evolutionary Algorithm for Solving U-Shaped Assembly Line Balancing Problem with Man-Robot Cooperation [J]. *Journal of Mechanical Engineering*, 2020, 56(9): 199-214.
- [27] Ishibuchi Hisao, Yoshida Tadashi, Murata Tadahiko. Balance Between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flowshop Scheduling[J]. *IEEE Transactions on Evolutionary Computation* (S1089-778X), 2003, 7(2): 204-223.