

DOI: 10.3901/JME.2020.09.199

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

郑逸凡¹ 钱斌^{1,2} 胡蓉^{1,2} 张长胜¹ 向凤红¹

(1. 昆明理工大学信息工程与自动化学院 昆明 650500;

2. 昆明理工大学云南省人工智能重点实验室 昆明 650500)

摘要: 针对一类广泛存在的生产装配问题, 建立人机共同作业的资源约束 U 形装配线平衡问题(Resource constraint U-shaped assembly line balancing problem with man-robot cooperation, RCUALBP_MRC)模型。该模型中机器人与助理均为有限资源, 机器人可替代人工操作, 助理可协助工人操作, 优化目标为同时最小化总成本指标和最大化线效率以及负载标准差综合指标。一种用于求解 RCUALBP_MRC 的基于交叉熵(Cross-entropy, CE)方法与遗传算法(Genetic algorithm, GA)的协同进化算法(CE-GA Co-evolutionary algorithm, CE-GACEA)被提出。首先, 根据问题特点, 对解中工序子序列设计了一种基于工序选择因子的编码(Task selection factor based code, TSFBC)。其次, 在算法的全局搜索阶段对解中工序子序列和机器人及助理子序列所确定的子空间, 分别利用 GA 和 CE 的操作进行协同搜索, 可丰富搜索方向并发现优质解区域; 局部搜索阶段加入种群分裂-合并机制, 可有效平衡算法的全局与局部搜索, 改善算法性能。最后, 通过在不同规模问题上的仿真试验和算法对比, 验证所提 CE-GACEA 的有效性。

关键词: U 形装配线平衡; 遗传算法; 交叉熵方法; 协同进化; 多目标优化; 人机共同作业

中图分类号: TG156

CE-GA Co-evolutionary Algorithm for Solving U-shaped Assembly Line Balancing Problem with Man-robot Cooperation

ZHENG Yifan¹ QIAN Bin^{1,2} HU Rong^{1,2} ZHANG Changsheng¹ XIANG Fenghong¹

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

2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and
Technology, Kunming 650500)

Abstract: Aiming at a kind of widely existing production assembly problem, the model of the resource-constraint U-shaped assembly line balancing problem with man-robot cooperation (RCUALBP_MRC) is built. In this model, robots and assistants are limited resources, robots can replace manual operations, and assistants can assist workers in operations. The criteria are to simultaneously minimize the objective of total cost as well as maximize the integrated objective of line efficiency and load variance. Based on the cross-entropy (CE) method and genetic algorithm (GA), a co-evolutionary algorithm (CE-GACEA) is proposed for solving the RCUALBP_MRC. Firstly, according to the characteristics of problem solution, an efficient coding called the "task selection factor based code" (TSFBC) is designed for the task subsequence in solution. Secondly, in the global search phase, the operations of GA and CE are used to collaboratively search the subspaces determined by the task subsequence as well as the robot and assistant subsequence in solution, which can enrich search directions and find promising regions. In the local search phase, the split-merge mechanism of population is adopted, which effectively balances the global and local search of the algorithm and improves the performance of the algorithm. Finally, simulation experiments and comparisons on different instances demonstrate the effectiveness of proposed algorithm.

Key words: U-shaped assembly line balancing; genetic algorithm; cross-entropy method; co-evolution; multi-objective optimization; man-robot cooperation

* 国家自然科学基金(51665025, 61963022)资助项目。20190919 收到初稿,
20191226 收到修改稿

0 前言

装配流水线是一种特殊的生产系统,将一件产品按照特定的工艺流程,以规定的速度顺序通过每个工位,依次完成所有装配工序,具备这样功能的生产系统就可以称之为装配线系统。装配线平衡问题(Assembly line balancing problem, ALBP)是制造领域的一个重要问题^[1],于1955年由 SALVESON^[2]首次提出,主要研究如何合理分配一系列具有操作优先约束和独立操作时间的工序至不同工位,以提升装配线系统的生产效率,该问题已被证明是一类典型的 NP-hard 组合优化问题^[3]。MILTENBURG 等^[4]于1994年首次提出 U 形装配线平衡问题(U-shaped assembly line balancing problem, UALBP)的数学模型,此后 MILTENBURG^[5]通过调研美国和日本 114 条装配生产线得出结论:对比没有采用 U 形布局的装配线,U 形装配线的生产效率平均提高 76%,同时产品缺陷率平均降低 83%。U 形装配线在效率、柔性和空间利用等方面具有独特优势,已被广泛应用于国内外汽车^[6]、飞机^[7]、电力设备^[8]等制造企业的生产线上,是制造业中一类重要的生产方式。显然,研究 UALBP 是缩短产品制造周期,从而提高我国制造企业竞争力的有效途径。

按照已知变量与所求未知量的不同,UALBP 可分为以下三类问题:① 已知工位节拍时间最小化工位数量的第一类问题(UALBP-1);② 已知工位数量最小化节拍时间的第二类问题(UALBP-2);③ 已知工位数量和节拍时间均衡各工位间负载的第三类问题(UALBP-3)。近年来国内外对 UALBP 已经开展了较多的研究,对各类 UALBP 的研究方法可大致分为精确方法和近似方法。由于 UALBP 属于 NP-hard 问题,复杂度随问题规模增加呈几何倍数增长,采用分支定界^[9]或数学规划^[10-11]等精确方法难以求解较大规模问题。因此一些学者采用启发式方法或智能优化算法等近似方法以求较短时间内获得问题的近似最优解。智能优化算法在求解调度问题时一般先对问题建立排序模型,然后通过某种机制引导序列向目标值最优的方向优化,建立排序模型的过程对应智能算法的编码和解码的设计。UALBP 存在工序操作优先约束和工位节拍约束,因此设计编码时需要确保算法生成的解是可行的,即工序分配方案满足上述的约束关系。目前对 UALBP 编码解码的研究有几种主流的方法:WAINWRIGHT 等^[12]对工序序号编码,通过算法初始化生成可行解

和采用“有序两点交叉”算子的方式保证后代解的可行性。该方法的不足之处在于,若加入变异或局部扰动算子会破坏原本满足约束的编码序列结构,算法将无法通过扰动操作改善自身性能。BAYKASOĞLU 等^[13]提出了十种工序分配规则,工序的每一次分配按照对应的规则进行,编码序列实质是规则的排序。多规则的操作机制能够在加入扰动操作的同时避免不可行解的产生,增强了算法跳出局部最优的能力,因此基于多规则的编码(Multiple-rules based code, MRBC)在 UALBP 研究中得到广泛应用^[13-16]。譬如,对于混流 UALBP-1, RABBANI 等^[14]采用基于多规则编码的遗传算法求解,优化目标为同时考虑线效率和工位负载标准差的综合指标。对于考虑工人与工序分配协同优化的 UALBP-2, ZHANG 等^[15]设计了基于多规则编码的增强候鸟优化算法求解,优化目标为最小化节拍时间。MRBC 编码可有效求解 UALBP,但由于需要对多种规则进行计算导致该方法的解码复杂度相对较高。对于 UALBP 的编码解码效率存在进一步改善和提升的空间。

随着准时制(Just in time, JIT)生产模式在制造行业的推广,如何合理分配有限资源以减少产能浪费,缩短生产周期日益受到重视。对于需要考虑装配线上的人力,设备等资源合理分配的 UALBP 近年来越来越受到国内外研究者的关注。OKSUZA 等^[17]采用人工蜂群算法求解考虑不同操作水平的工人以及工序分配的 UALBP-1,优化目标为最大化线效率。KARA 等^[18]首次提出考虑操作时间依赖设备与助理资源的 UALBP(Resource dependent UALBP, RDUALBP),其中设备和助理可辅助普通工人操作以缩短操作时间,同时建立了以最小化总成本为目标的数学规划模型并采用“Gurobi LP/MIP Solver Engine”求解。JAYASWAL 等^[19]在文献[18]工作的基础上建立这类 RDUALBP 的改进模型,并设计模拟退火算法求解。张子凯等^[16]将文献[18-19]研究的单目标 RDUALBP 扩展为同时考虑线效率和总成本的双目标优化问题,并提出一种基于 BENDERS 分解的快速非支配遗传算法求解。上述文献中的 RDUALBP 只考虑了设备与助理辅助工人进行操作,而未考虑机器人参与生产的情况。近年来,机器人逐渐在部分工位上替代人工进行操作。随着机器人和人工智能技术的快速融合和发展,这一趋势越来越明显,必将引发国内外制造业的深刻变革。同时,我国在《中国制造 2025》中明确指出,在实行五大工程之一的“智能制造工程”时,需要“紧

扣关键工序智能化、关键岗位机器人替代、生产过程智能优化控制、供应链优化,建设重点领域智能工厂/数字化车间”^[20]。其中,“关键岗位机器人替代”是实现智能工厂的重要环节。然而受限于目前机器人技术的发展以及较高的投入成本,很多制造企业普遍采用人工与机器人共同作业的生产方式。因此,对人机共同作业的资源约束 U 形装配线平衡问题(Resource constraint u-shaped assembly line balancing problem with man-robot cooperation, RCUALBP_MRC)这类 NP-hard 问题开展相关研究,不仅具有重要的理论和现实意义,也符合国家战略。然而,目前尚无 RCUALBP_MRC 的研究报道,故迫切需要开展相关研究。

交叉熵(Cross-entropy, CE)方法的概念最早由 RUBINSTEIN^[21]提出用于估计复杂随机网络中稀有事件发生的概率,其主要思想是将优化问题和与其相关的概率模型估计问题相关联,通过更新稀有事件对应的概率模型参数并使用有效采样方法采样概率模型以生成新种群来引导搜索方向,使算法逐渐逼近最优解或次优解。已有部分学者将 CE 方法应用到组合优化问题的求解中。CASERTA 等^[22]设计了基于 CE 的元启发式算法求解整数背包问题,经过在大规模实例上的对比试验,验证其性能优于分支定界法。CHEPURI 等^[23]采用 CE 方法求解车辆路径问题,对不同优化目标和约束下的问题都取得了较好的效果。WANG 等^[24]采用一种改进的 CE 方法求解复杂炼钢-连铸生产调度问题并通过试验对比验证了其性能优于传统 CE 方法。CE 方法自适应和鲁棒性强,具有较好的全局搜索能力,但也存在搜索行为单一所导致的容易过早收敛的不足。因此有研究者为了改善 CE 方法的性能将该方法与传统智能优化算法以及局部搜索操作相结合。譬如, JĘDRZEJOWICZ 等^[25]设计了一种融合 CE 的人群学习算法求解离散资源约束下的并行机调度问题,该算法通过加入禁忌搜索算法以增强其局部搜索能力。BUDI 等^[26]提出一种混合交叉熵遗传算法求解以最小化最大完工时间为目标的零等待作业车间调度问题,该算法在交叉熵算法基础上,进一步利用遗传算法的交叉、变异操作来丰富算法的搜索行为,使算法可同时对解空间不同区域进行搜索。李国成等^[27]将 CE 方法与蝙蝠优化算法相结合,提出一种交叉熵蝙蝠算法求解高维函数优化问题,该算法采用 CE 方法和蝙蝠算法两套不同搜索机制进行协同搜索,以提高算法对解空间不同区域的搜索能力。

协同进化算法(Co-evolutionary algorithm, CEA)

源于自然界和社会系统广泛存在的合作、竞争、共生等现象,通过不同层面的协同搜索,可丰富搜索行为,从而有效改善单个进化算法的性能,因此逐渐成为进化计算领域研究的热点^[28]。CEA 已在组合优化问题的求解中得到了广泛应用。对于柔性作业车间调度问题, LEI^[29]设计一种协同进化遗传算法,对工序和机器序列采用了两种不同的交叉算子,优化目标为最小化模糊完工时间。WANG 等^[30]提出一种双种群协同的分布估计算法,分别采用两个概率模型优化工件序列和机器序列,优化目标为最小化最大完工时间。对于混流 UALBP,刘冉等^[31]提出了一种基于 Pareto 支配的多目标克隆免疫协同进化算法,以两个单克隆抗体群分别对平衡与调度两个子问题进行编码并协同进化。KIM 等^[32]提出一种基于种群内部共生机制的协同进化算法,采用选择共生伙伴的方法在群体内部和群体之间进行局部互动,以协同优化平衡和排序两个子问题。现有研究表明,协同进化算法能更有效地增强种群多样性,并丰富搜索行为,从而可引导算法搜索问题解空间中更多的不同区域,这有利于提高算法性能。

本文对一类人机共同作业的资源约束 U 形装配线平衡问题(RCUALBP_MRC)进行建模和求解算法研究。在 RCUALBP_MRC 中,机器人和助理都为有限资源,自动化机器人可以替代工人操作,助理定义为高级技工,可以协助普通工人操作,工序操作时间与工人、机器人和助理的分配相关,优化目标为最大化线效率与工位负载标准差综合指标和最小化总成本指标的双目标。RCUALBP_MRC 是 NP-hard 问题,故其解空间巨大且复杂,优质解不规则地分布在不同区域。这表明单一搜索机制难以有效搜索解空间。CE 方法通过采样概率模型生成新种群,而遗传算法利用选择、交叉、变异操作产生新种群,两者分别属于进化算法中两类完全不同的搜索机制。利用 CE 方法和遗传算法协同进化,可丰富算法的搜索行为,有利于搜索解空间中更多的不同区域,从而能在较短时间内找到问题的优质解或近似最优解。此外,对于 RCUALBP_MRC,将问题的解分解为前后两段,并由遗传算法和 CE 方法分别对前段和后段对应的子空间进行搜索,有利于控制搜索区域,提高搜索效率。因此,采用 CE 方法和遗传算法构造协同进化算法,有望使得混合后算法的搜索覆盖较多存在优质解的区域,进而可成为求解 RCUALBP_MRC 的有效算法。显然,开展相关研究具有重要意义。

针对 RCUALBP_MRC,本文提出一种基于交

叉熵(Cross-entropy, CE)方法与遗传算法(Genetic algorithm, GA)的协同进化算法(CE-GA co-evolutionary algorithm, CE-GACEA)进行求解。首先,在算法编码和解码阶段,设计问题解的工序、机器人和助理三段式编码和解码机制,同时根据问题结构特点,为工序编码序列构建一种基于工序选择因子的编码(Task selection factor based code, TSFBC)。TSFBC 不仅能降低解码复杂度,也可确保对解或序列执行合法扰动。其次,在算法全局搜索阶段,设计一种 CE-GA 协同进化机制,将 GA 和 CE 方法分别用于对问题解的前段(对应工序分配)和后段(对应机器人和助理分配)进行搜索。已有的协同进化算法常将同一算法的种群分裂为多个子种群并分别用于求解相应子问题,或者将不同算法交替用于求解整个问题。本文采用不同算法共同对问题解进行分段搜索,可提高搜索不同区域的机率。然后,在算法的局部搜索阶段加入种群分裂-合并机制,以进一步增强算法的局部搜索能力。通过分裂操作产生两个子种群,进而对子种群 1(子种群 2)中个体的前段(后段)进行变邻域搜索并保持后段(前段)不变,可对解空间特定区域执行更深入的搜索。通过合并操作使子种群中的优质解或个体实现相互迁移,从而确保子种群对不同搜索区域的较均匀覆盖,有效地平衡算法的全局与局部搜索。最后,仿真试验和算法对比验证了 CE-GACEA 是求解 RCUALBP_MRC 的有效算法。

1 RCUALBP_MRC 的问题描述

相较于传统直线型装配线, U 形装配线上的工位同时包含了产品的入口线和出口线,如图 1 所示。设装配线工位的节拍时间为 10(单位时间),相同工序分配方案下采用 U 形布局的装配线比采用直线布局的装配线需要的工位数量更少,实际工位操作时间也更为接近节拍时间,由此可见 U 形装配线具备更高的生产效率和柔性。

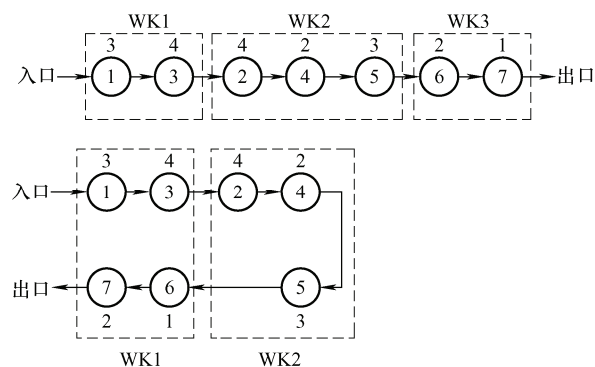


图1 U形装配线与直线型装配线

人机共同作业广泛存在于汽车发动机的装配中。传统的工业机器人已可代替人工完成部分高强度但对操作精度与灵活性要求不高的工作(例如焊接、起重、喷涂等)。近年来一些具有高自由度与高精度的机器人开始被应用于发动机装配,例如图 2 所示的多关节机器臂可一次对多个指定部位进行螺丝拧紧。现以重庆某汽车企业的发动机装配线为例描述 RCUALBP_MRC, 该装配线为 U 形布局, 不同工位按实际需求分配机器人、工人和助理进行操作, 其中由机器人完成缸体翻转、编号打印、松卸瓦盖, 安装上下瓦片等 13 道工序; 由工人完成进气管和进油管安装、外观检查, 油系试漏等 14 道工序。机器人只能被固定在工位的一侧, 而工人和助理可以在工位两侧移动, 具体如图 3 所示。同时, 该装配线所包含 27 个工序的先后约束如图 4 所示。

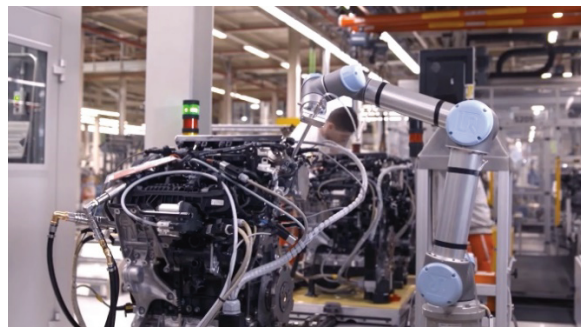


图2 机器人在发动机装配中的应用

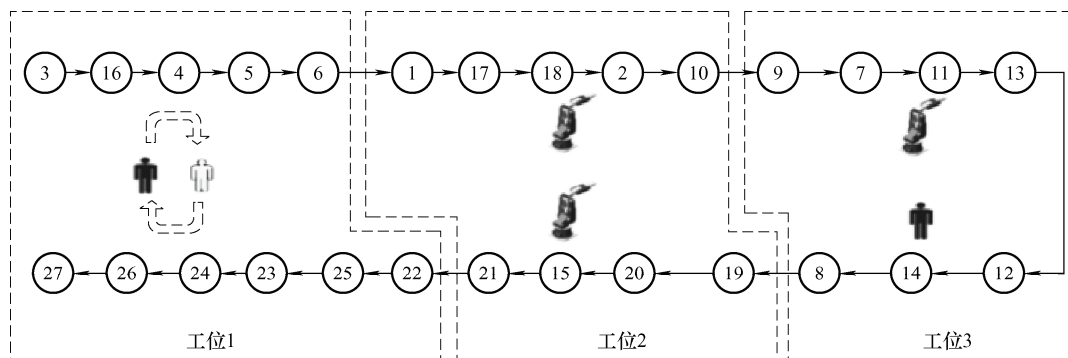


图3 装配线各工位的资源分配情况

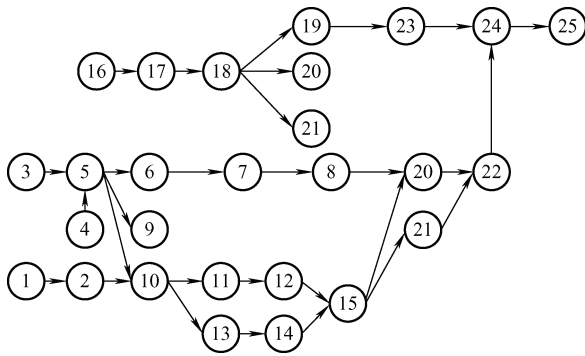


图4 工序间的约束关系

1.1 问题假设

为便于研究,对 RCUALBP_MRC 做出以下假设。

- (1) 本文考虑的装配线只生产单一产品。
- (2) 工人和助理可以在工位的两侧移动完成入口线和出口线的工序操作,不考虑工人移动的时间。
- (3) 工序确定的操作时间为只有工人进行操作的时间。

(4) 机器人只能被安排到工位的出口线或入口线一侧,不能同时操作工位两侧的工序。

(5) 工人、助理和机器人可操作多个工序,但同一工序不可被工人(或助理)与机器人同时操作。

(6) 助理和机器人资源是有限的。

1.2 符号定义

RCUALBP_MRC 的符号定义可由表 1 给出。

1.3 数学模型

RCUALBP_MRC 的数学模型可表示如下

$$\max \left\{ \frac{\sum_{i \in I} t_i}{CT \times \sum_{j \in J} j} + \left[1 - \sqrt{\frac{1}{\sum_{j \in J} j} \times \sum_{j \in J} (u_j - a)^2} \right] \right\} \quad (1)$$

$$\min \sum_{j \in J} CS_j \quad (2)$$

$$\sum_{j \in J} (X_{ij} + Y_{ij}) = 1 \quad \forall i \in I \quad (3)$$

$$\sum_{j \in J} (X_{ij} - X_{\bar{i}j}) + (Y_{ij} - Y_{\bar{i}j}) \geq 0 \quad \forall (\tilde{i}, \bar{i}) \in PR \quad (4)$$

$$t_i = \begin{cases} t_{i0} & RbX_{rj} \times X_{ij} + RbY_{rj} \times Y_{ij} = 0 \& \\ & A_j W_j (X_{ij} + Y_{ij}) = 0 \\ t_{ir} & RbX_{rj} \times X_{ij} + RbY_{rj} \times Y_{ij} \neq 0 \quad \forall j \in J, \forall r \in R \\ t_{ia} & RbX_{rj} \times X_{ij} + RbY_{rj} \times Y_{ij} = 0 \& \\ & A_j W_j (X_{ij} + Y_{ij}) \neq 0 \end{cases} \quad (5)$$

$$W_j = \begin{cases} 1 & RbX_{rj} + RbY_{rj} \leq 1, \forall r \in R \\ 0 & RbX_{rj} + RbY_{rj} > 1, \forall r \in R \end{cases} \quad \forall j \in J \quad (6)$$

表1 相关符号定义

符号	说明
I	工序集合, $I = \{1, 2, \dots, i\}, i \in I$;
J	工位集合, $J = \{1, 2, \dots, j\}, j \in J$;
PR	优先约束关系集合, $(\tilde{i}, \bar{i}) \in PR$ 表示 \tilde{i} 是 \bar{i} 的直接前序, 即工序 \bar{i} 只能在工序 \tilde{i} 完成后开始;
R	机器人类型集合, $R = \{1, 2, \dots, r\}, r \in R$;
t_i	工序 i 的操作时间, $i \in I$;
t_{i0}	只有工人操作时工序 i 的操作时间;
t_{ir}	工序 i 使用机器人 r 的操作时间, $i \in I, r \in R$;
t_{ia}	工序 i 在助理协助下的操作时间;
CT	节拍时间;
NR_r	机器人 r 可使用的数量;
NR	所有可使用机器人的数量;
NA	可使用助理的数量;
CS_j	工位 j 的总成本;
CW	使用一个工人的成本;
CR_r	使用机器人 r 的成本;
CA	使用一个助理的成本;
CJ	开设一个工位的成本;
X_{ij}, Y_{ij}	工序 i 是否分配在工位 j 的入口线或出口线, $X_{ij}, Y_{ij} \in (0, 1)$;
RbX_{rj}, RbY_{rj}	表示机器人 r 是否分配在工位 j 的入口线或出口线, $RbX_{rj}, RbY_{rj} \in (0, 1)$;
W_j	表示工位 j 能否分配工人(或助理), $W_j \in (0, 1)$;
A_j	表示工位 j 是否分配助理, $A_j \in (0, 1)$;

$$\sum_{i \in I} t_i (X_{ij} + Y_{ij}) \leq CT, \forall j \in J \quad (7)$$

$$\sum_{j \in J} (RbX_{rj} + RbY_{rj}) \leq NR, \forall r \in R \quad (8)$$

$$\sum_{j \in J} A_j W_j \leq NA \quad (9)$$

其中, 式(1)和式(2)为 RCUALBP_MRC 的目标函数。式(1)是 RABBANI 等^[10]提出的线效率与工位负载标准差的综合指标。线效率定义为实际总操作时间与规定节拍时间的比值。负载标准差定义为各工位实际效率的标准差, 值小说明装配线的稳定性和抗干扰能力越强。工位实际效率 u_j 与工位平均效率 a 的计算分别由式(10)和(11)给出。

$$u_j = \frac{1}{CT} \times \sum_{i \in I} t_i (X_{ij} + Y_{ij}), \forall j \in J \quad (10)$$

$$a = \frac{1}{\sum_{j \in J} j} \times \sum_{j \in J} u_j \quad (11)$$

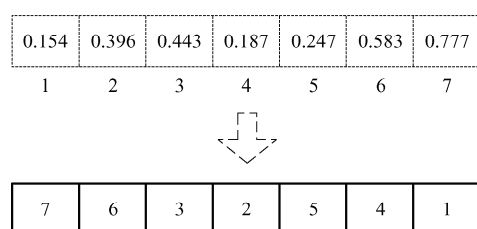
式(2)是最小化总成本的表达式, 每个工位成本的计算由式(12)给出

$$CS_j = CW \times W_j + CA \times W_j + CR_r \times (RbX_{rj} + RbY_{rj}) + CJ, \forall j \in J, \forall r \in R \quad (12)$$

式(3)至式(9)为约束条件表达式。其中,式(3)表示每个工序都要被分配,且只能分配到工位的一侧;式(4)是工序分别在工位的入口线或出口线上需要满足的优先约束关系;式(5)规定了工序的实际操作时间,取决于工序当前由机器人、工人或助理进行操作;式(6)中 W_j 为0-1变量,当工位两侧的工序都由机器人操作时 W_j 为0,表示此时工位不能分配工人和助理;式(7)为节拍约束,节拍时间定义为一个生产周期内每个工位规定的最大操作时间,实际的工位操作时间不能大于的节拍时间;式(8)和式(9)分别为机器人和助理的数量约束,均为有限资源。

2 CE-GA 协同进化算法(CE-GACEA)

协同进化的基本思想是对复杂问题的模块化处理与整合,即通过不同层面的协同操作来丰富算法寻优的方向^[28]。本文所采用的 CE-GA 协同进化算法(CE-GACEA)以不同子序列为模块,从个体及种群两个层面实现协同进化。首先,在算法的全局搜索阶段分别采用遗传算法(下文称 GA)的操作生成同一个体的工序分配子序列,交叉熵方法(下文称 CE)的操作生成机器人和助理分配子序列;其次,在局部搜索阶段对分裂为两个规模相同的子种群执行对特定子序列的邻域搜索操作;然后通过特定条件下的分裂与合并机制实现问题模块的拆分与整合。



2.1 编码和解码

2.1.1 基于工序选择因子的编码

部分采用智能优化算法求解 UALBP 的文献中,工序编码存在无法对序列执行扰动操作^[12]或解码过程计算较为复杂^[13]等问题。为了改善这些问题并提高编解码的效率,本文对 UALBP 的工序分配问题采用一种基于工序选择因子的编码 (Task Selection Factor Based Code, TSFBC)。TSFBC 编码的核心思想在于将每一次工序分配的选择优先级信息直接与工序关联,从而减少计算复杂度,提高编解码的效率。

TSFBC 的原理如图 5 所示。首先,算法初始化为每个工序生成 0 到 1 之间的不重复随机实数,工序分配时对于所有满足约束关系的候选工序,将选择对应随机实数最大的放入当前工位,待所有工序分配完成后将每个工序的序号按对应随机实数从大到小排序得到工序分配子序列。然后,算法运行时先生成一组 0 到 1 之间,规模与工序数相同并被从大到小排序的不重复随机实数,这组实数被称为选择因子。选择因子在算法优化过程中将按照工序子序列对应匹配到每个工序,同样的工序在进行分配时,所有满足约束关系的工序中具有最大选择因子的将被选择放入当前工位。该编码中选择因子的值实际为工序分配优先关系的映射,算法对序列的优化本质上是获得更优工序分配优先关系的过程。

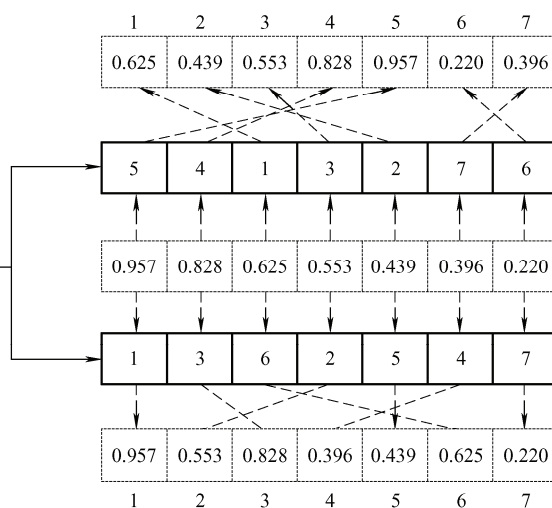


图 5 TSFBC 编码示例

2.1.2 工序,机器人和助理的三段式编码

由于 RCUALBP_MRC 属于已知节拍时间而工位数量未知的 UALBP-1,为使编码序列能表示所有工位上机器人和助理的分配情况,按照张子凯等^[16]的方法规定最大工位数 M 为其下限值的两倍

$$M = 2 \times \left(\sum_{i \in I} t_i / CT \right) \quad (13)$$

工序,机器人和助理的三段式编码如图 6 所示。工序编码为工序选择因子的值从大到小的排序,长度 $N_i = \sum_{i \in I} i$ 。机器人编码从编码序列的第 $N_i + 1$ 位开始,长度为 $2M$,每两位表示一个工位机器人分配的情况,前一位表示工位的入口线,后一位表示工位的出口线,若取值为 0 表示不分配机器人,取

非 0 值则表示分配机器人的类型。例如图 6 所示的机器人分配中, 工位 1 的入口线不分配机器人, 出口线分配机器人 1。助理编码从编码序列的第 $N_i + 2M + 1$ 位开始, 长度为 M , 每一位表示工位使用助理的情况, 取值 0 或 1 分别表示对该工位是否分配助理。例如图 6 所示的助理分配中, 工位 1 分配助理, 工位 2 和工位 3 不分配助理。

							$j=1$		$j=2$		$j=3$		$j=1$		$j=2$		$j=3$	
7	6	3	2	5	4	1	0	1	3	1	1	2	1	0	0	0	0	0
工序							机器人						助理					

图 6 工序, 机器人和助理三段式编码

2.1.3 解码

具体的解码步骤如下所述。

(1) 初始化当前工位 $k=1$, 当前工序数 $l = \sum_{i \in I} i$

可用助理数量 $num_A = NA$, 可用机器人数量 $num_R = NR$, 可选择工序集 $D = I$, 工位操作时间 $t_k = 0$, 按工序编码分配选择因子至各工序。

(2) 构造可分配工序集 D_j , 入口线工序集 D_{in} 和出口线工序集 D_{out} 都为空, 对所有满足可分配条件的工序 i , 执行 $D_j = D_j + i$, $D_{in} = D_{in} + i$ ($D_{out} = D_{out} + i$)。

(3) 选择 D_j 中具有最大选择因子的工序 i' , 执行 $D = D - i'$, D_j 置为空集。

(4) 判断机器人和助理是否可用, 若 $num_R = 0$, 则将序列中表示 k 工位的机器人编码位置 0; 若 $num_A = 0$, 则将序列中表示 k 工位的助理编码位置 0。

(5) 若序列中表示 k 工位入口线和出口线的机器人编码都不为 0, 使表示 k 工位机器人数量的变量 $num_R_k = 2$, 转步骤(6); 若入口线或出口线机器人编码其中一位为 0, 使 $num_R_k = 1$, 转步骤(7); 若入口线和出口线机器人编码都为 0, 使 $num_R_k = 0$, 转步骤(8)。

(6) 根据被选择的工序 $i' \in D_{in}$ 或 $i' \in D_{out}$, 计算工序 i' 的操作时间 $t_i = t_{ir}$, 转步骤(9)。

(7) 根据被选择的工序 $i' \in D_{in}$ 或 $i' \in D_{out}$ 判断对应入口线(或出口线)上是否分配机器人, 若有机器人分配则计算工序 i' 的操作时间 $t_i = t_{ir}$, 转步骤(9); 若未分配机器人则转步骤(8)。

(8) 若序列中表示 k 工位助理分配的编码为 1, 使 $num_A_k = 1$, 计算工序 i' 的操作时间 $t_i = t_{ia}$; 若为 0, 使 $num_A_k = 0$, $t_i = t_{i0}$ 。

(9) 令 $t_k = t_k + t_{i'}$, 若满足 $t_k \leq CT$, 转步骤(10); 若 $t_k > CT$, 则执行 $num_A = num_A - num_A_k$,

$num_R = num_R - num_R_k$, $t_k = t_k - t_{i'}$, $k = k + 1$, $t_k = 0$, $D = D + i'$, 转步骤(2)。

(10) $l = l - 1$, 若 $l > 0$ 则继续执行步骤(2)至步骤(9), 若 $l = 0$ 结束。

2.2 多目标问题的 Pareto 支配法

RCUALBP_MRC 为同时考虑线效率, 负载标准差及成本的多目标问题, 现有解决多目标问题的研究方法大致分为两类: 目标加权法和 Pareto 支配法。目标加权法由于受到不同目标主观权重的影响不能客观的反映解的优劣, 因此本文采用 Pareto 支配法处理多目标问题。

多目标优化可以描述为

$$\min f(x) = (f_1(x), f_2(x), \dots, f_k(x)), \quad \forall x \in X \quad (14)$$

式中, x 为决策变量; X 为可行解集合; $f(x)$ 为目标函数; k 为优化目标的数量。

给出 Pareto 多目标优化的几个基本概念如下。

(1) Pareto 支配: 对于解 x_1 和 x_2 , 若解 x_1 能支配 x_2 , 当且仅当 $\forall i \in \{1, 2, \dots, K\}$, $f_i(x_1) \leq f_i(x_2)$, 同时 $\exists j \in \{1, 2, \dots, K\}$, 使得 $f_j(x_1) < f_j(x_2)$ 。

(2) 非支配解集: 解集中不能被其他解支配的解所构成的集合。

(3) 非支配排序: 根据支配关系将解集中的解分为不同支配等级的过程。第 1 级包括解集中所有非支配解, 第 2 级中包括仅被第 1 级中的解支配的解, 以此类推, 解 x 的支配等级记为 $rank(x)$ 。

(4) 拥挤距离: 用于评价同一支配等级中解之间的分散程度, 解 x 的拥挤距离记为 $D(x)$, 通常若 $rank(x_1) < rank(x_2)$ 或 $rank(x_1) = rank(x_2) \& D(x_1) > D(x_2)$ 则认为解 x_1 在 Pareto 意义下优于 x_2 。给出解 x 拥挤距离的计算

$$D(x) = D_1(x) + D_2(x) + \dots + D_k(x) \quad (15)$$

$$D_k(x) = [D_k(x+1) - D_k(x-1)] / [f_k^{\max} - f_k^{\min}] \quad (16)$$

其中, $D_k(x+1)$ 与 $D_k(x-1)$ 分别是解 x 在目标 k 下排列的两个相邻解, f_k^{\max} 和 f_k^{\min} 是目标函数 k 在当前支配等级下的最大值和最小值。

(5) Pareto 档案集: 在整个解空间中不被任何解支配的解构成的集合。

(6) Pareto 前沿: 所有 Pareto 最优解在目标空间呈现的形状。

2.3 CE-GA 协同的全局搜索

2.3.1 工序分配子序列的生成

采用 GA 的操作生成工序分配子序列。GA 中包含三个主要的算子: 选择, 交叉, 变异。选择是从当前种群中按个体适应度为标准, 以一定概率选择优良个体作为父代的操作, 在多目标优化问题中

GA 通常比较个体所在层级以及拥挤距离作为选择的依据^[33],也有算法对两个目标进行加权处理得到一个虚拟适应度^[34]。本文在选择阶段采用二元锦标赛的方法,每次在种群中随机选择两个个体,比较他们的非支配层级和拥挤距离,选择非支配层级较小的或在相同非支配层级的条件下选择拥挤距离较大的个体作为父代个体。

交叉算子将父代优秀的基因特征遗传给子代。本文的交叉操作采用两点交叉的方式,随机决定父代 1 的两个点位,将第一个点位之前和第二个点位之后的工序及对应位置信息全部保留到子代,然后从父代 2 选择当前子代缺失的工序按照父代 2 中的顺序保留到子代。

为了保证种群的多样性,改善解容易陷入局部最优的情况,GA 中还引入了变异算子。本文采用一种随机自适应变异的机制,个体的变异概率 P_m 与迭代的时间有关,式(17)表示当迭代时间 t 接近最大迭代时间 T_{\max} 需要适当增加扰动使个体容易跳出局部最优,先对 P_m 赋予一个 0 到 1 之间的随机概率,式(17)为个体发生变异的条件,变异操作为前向插入

$$P_m \leq 0.5 \times \frac{t}{T_{\max}} \quad (17)$$

本文 GA 中所采用的交叉和变异操作如图 7 所示。

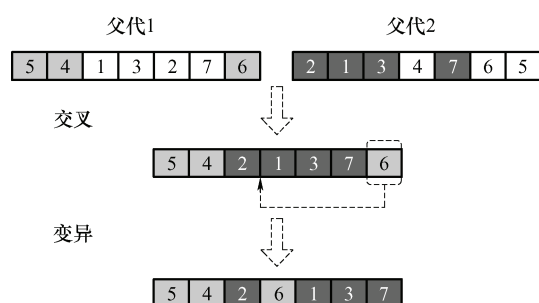


图 7 交叉和变异操作

2.3.2 机器人和助理分配子序列的生成

采用 CE 的操作生成机器人和助理分配子序列。CE 的两个关键步骤在于随机样本的抽取和概率参数的确定。令 $X = (x_1, x_2, \dots, x_n)$ 是某个概率空间 χ 的一个随机变量,本问题中该随机向量表示的就是工位的入口线和出口线机器人(或助理)分配情况的一组调度方案。以机器人分配为例,随机变量 $x_i (i=1, 3, \dots, n-1)$ 表示工位 $(i+1)/2$ 的入口线按概率 $P_i = (p_{i1}, p_{i2}, \dots, p_{ik})$ 在机器人的 k 个可选取值中选择一个值;同样的,随机变量 $x_i (i=2, 4, \dots, n)$ 表示工位 $i/2$ 的出口线按概率 $P_i = (p_{i1}, p_{i2}, \dots, p_{ik})$ 在机器人的 k 个可选取值中选择一个值,概率 P_i 均

满足 $\sum_{i=1}^k p_{il} = 1$ 。

下一步进行概率参数的确定,1.3 节中问题的目标函数可等价表示如下

$$\begin{cases} S_1(X^*) = \gamma_1^* = \max_{X \in \chi} S_1(X) \\ S_2(X^*) = \gamma_2^* = \min_{X \in \chi} S_2(X) \end{cases} \quad (18)$$

式中, χ 是所有机器人分配方案的解空间, X^* 是最优调度方案, γ_1^* 和 γ_2^* 分别是由目标约束 S_1 和 S_2 确定的最优值。令样本 X 需要服从概率参数为 μ 的概率分布 $f(\cdot; \mu)$, (18)中求解最优 X^* 和 γ^* 的问题可转化为如何确定最优概率分布 $f(X; \mu)$ 的一个概率估计问题

$$\begin{cases} l(\gamma_1) = P_\mu(S_1(X) \geq \gamma_1) = \\ \sum_X I_{\{S_1(X) \geq \gamma_1\}} \cdot f(X; \mu) = E_\mu I_{\{S_1(X) \geq \gamma_1\}} \\ l(\gamma_2) = P_\mu(S_2(X) \leq \gamma_2) = \\ \sum_X I_{\{S_2(X) \leq \gamma_2\}} \cdot f(X; \mu) = E_\mu I_{\{S_2(X) \leq \gamma_2\}} \end{cases} \quad (19)$$

式中, P_μ 和 E_μ 分别是对分布 $f(\cdot; \mu)$ 的概率度量和期望;对于 $\gamma_1(\gamma_2)$, $I_{\{S_1(X) \geq \gamma_1\}}(I_{\{S_2(X) \leq \gamma_2\}})$ 是对于 χ 上不同 γ 值的示性函数集合, $E_\mu I_{\{S_1(X) \geq \gamma_1\}}(E_\mu I_{\{S_2(X) \leq \gamma_2\}})$ 表示样本中目标优于 γ_1 的期望值。以 $l(\gamma_1)$ 为例,对 l 的估计采用重要抽样的方法,根据 χ 上可以使 $S(X)$ 接近 γ 的某分布 g 进行抽样,进而得到能使 l 方差为 0 的一个分布 g^* 。

$$\hat{l} = \frac{1}{N} \sum_{i=1}^N I_{\{S_1(X_i) \geq \gamma_1\}} \frac{f(X_i; \mu)}{g(X_i)} \quad (20)$$

$$g^*(X) = \frac{I_{\{S_1(X) \geq \gamma_1\}} f(X; \mu)}{l} \quad (21)$$

为估计 l 此时需要确定概率参数 μ 使分布 f 与分布 g^* 差距最小。衡量两个概率分布的 Kullback-Leibler(K-L)距离以判断他们之间的相似度,最小化 K-L 距离等价于最小化交叉熵,分布 p 对 q 的交叉熵定义为 q 分布的自信息对 p 分布的期望

$$H(p, q) = E_{x \sim p}[-\ln q(x)] = -\int p(x) \ln q(x) dx \quad (22)$$

此时问题变为求概率参数 μ 使 g^* 与 $f(X; \mu)$ 的交叉熵最小化的问题,代入(21)后可得

$$\begin{aligned} \mu^* &= \arg \min_{\mu} \left[-\int g^*(X) \ln f(X; \mu) dX \right] = \\ &\arg \max_{\mu} E_\mu I_{\{S_1(X) \geq \gamma_1\}} \ln f(X; \mu) \end{aligned} \quad (23)$$

由于样本 X 由概率变量 P_i 生成,概率分布 $f(X; \mu)$ 可由 P_i 等价表示如下

$$f(X; p) = \prod_{i=1}^n \prod_{l=1}^k p_{il}^{I_{\{x_i=l\}}} \quad (24)$$

其中, 示性函数 $I_{\{x_i=j\}}$ 表示机器人编码的第 i 位是否为 j , $\prod_{i=1}^n \prod_{l=1}^k p_{il}^{I_{\{x_i=l\}}}$ 表示的就是采用某一组机器人分配方案的概率。考虑概率 P_i 满足 $\sum_{l=1}^k p_{il} = 1$, 对式 (23) 使用拉格朗日乘子法求解

$$\max_p \min_{\omega_1, \dots, \omega_n} \left[E_p I_{\{S_i(X) \geq \gamma_i\}} \ln f(X; p) + \sum_{i=1}^n \omega_i \left(\sum_{l=1}^k p_{il} - 1 \right) \right] \quad (25)$$

对所有 $l=1, 2, \dots, k$, 有

$$\frac{E_p I_{\{S_i(X) \geq \gamma_i\}} I_{\{x_i=l\}}}{p_{il}} + \omega_i = 0 \quad (26)$$

令 $-\omega_i = E_p I_{\{S(X) \leq \gamma\}}$, 对于 m 个有效样本, 进行离散化处理后最优 p_{il} 可如下给出

$$p_{il} = \frac{\sum_{v=1}^m I_{\{S_i(X_v) \geq \gamma_i\}} I_{\{x_{vi}=l\}}}{\sum_{v=1}^m I_{\{S_i(X_v) \geq \gamma_i\}}} \quad (27)$$

式中, $\sum_{v=1}^m I_{\{S_i(X_v) \geq \gamma_i\}}$ 表示 m 个样本中目标值优于 γ 的样本个数, 示性函数 $I_{\{x_{vi}=l\}}$ 表示当前样本 v 中机器人编码 i 位是否选择机器人 l 。上述对机器人分配问题概率参数的确定方法对助理分配问题同样适用。

CE 通过采样式 (27) 所确定的概率参数生成机器人、助理分配子序列, 具体步骤如下所述。

(1) 第 k 次迭代中机器人(助理)编码的概率分布参数为 $p(k-1)$, 通过轮盘赌的采样方式随机生成一定数量的个体, 构成本次迭代的一组随机样本 $\{X_1, X_2, \dots, X_n\}$ 。

(2) 由步骤 (1) 中产生的随机样本按式 (27) 确定本次迭代的概率参数 $p(k)$ 。单目标优化问题中通过排序样本 $\{X_1, X_2, \dots, X_n\}$ 的目标函数值 $\{S(X_1), S(X_2), \dots, S(X_n)\}$, 计算样本的 ρ 分位数, 使得 $P(S(X) \leq \gamma_k^*) = \rho$; 多目标问题中令样本的 ρ 分位数为种群中 Pareto 意义下最优的前 $\rho\%$ 个体。

(3) 对于第 $k+1$ 次迭代机器人(助理)编码的概率参数, 引入平滑系数 α 的更新策略如下

$$p(k) = \alpha p(k) + (1 - \alpha) p(k-1) \quad (28)$$

2.4 基于种群分裂-合并机制的局部搜索

当算法在某一段时间内无法继续搜索到 Pareto 意义下更优的解, 则表示算法的全局搜索可能陷入

了局部最优, 此时需要进行局部搜索来增强算法跳出局部最优的能力。给出本文所采用的基于种群分裂-合并机制的局部搜索每个步骤具体操作如下。

(1) 种群分裂: 设定当 Pareto 档案集连续 η_g 代未更新时触发种群分裂的条件, 停止全局搜索操作并依次将种群规模为 num_s 的个体随机放入子种群 SP1 和 SP2, 子种群的规模相同均为 $num_s/2$ 。

(2) 对特定子序列的邻域搜索: 种群处于分裂状态时各子种群的所有搜索操作均只针对特定子序列进行: 对于 SP1 中的个体只对工序分配子序列操作, 机器人和助理分配子序列保持不变; 对于 SP2 中的个体只对机器人和助理分配子序列操作, 工序分配子序列保持不变。

SP1 采用随机组合的方式生成新个体, 如图 8 所示: 首先生成一系列长度等于工序子序列, 随机编码 0, 1, 2 的序列 T 。从 SP1 中随机选择两个个体, T 中编码为 1 的位置保留个体 1 中相同位置上的工序编码, 编码为 2 的位置按顺序保留个体 2 中未在新个体中出现过的工序编码, 编码为 0 的位置随机产生新个体中未出现的工序编码。当生成的新个体数量与原 SP1 个体数量之和达到 num_s , 对该种群所有个体进行非支配排序, 选出 Pareto 意义下最好的 $num_s/2$ 数量个体替换原 SP1。

SP2 采用随机交换的方式生成新个体, 如图 8 所示: 每次先随机生成一系列长度等于机器人(助理)子序列的 0, 1 序列 T , 从 SP2 中随机选择两个个体, 交换 T 中编码为 1 位置上两个个体对应的编码, 生成两个新的个体, 直到生成的新个体数量与原 SP2 个体数量之和为 num_s , 对该种群所有个体进行非支配排序, 选出 Pareto 意义下最好的 $num_s/2$ 数量个体替换原 SP2。

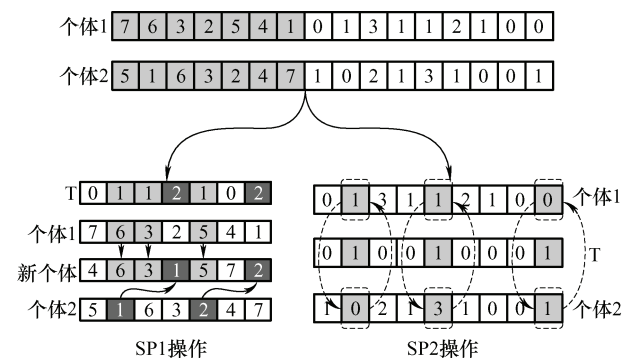


图 8 子种群的局部操作

为了进一步增强算法的寻优能力, 对 SP1 和 SP2 中的最优解执行基于三种邻域策略的变邻域搜索操作: Insert 称为邻域插入, 又分为前向 Insert 与后向 Insert, 随机选择序列的两个位置 p_1 和 p_2 ($p_1 > p_2$), 将 p_1 位的编码插入 p_2 位即前向

Insert, 反之则为后向 Insert; Interchange 称为邻域交换, 随机选择序列的一个位置 p , 交换 p 前后两个位置 p_1 和 p_2 上的编码。设定最大搜索次数为 5, 每次操作随机选择一种邻域(前向和后向 Insert 定义为两种不同的邻域操作), 当搜索到的新解可以支配旧解时下一次搜索沿用前一次邻域操作, 否则下一次搜索将改变邻域操作。

(3) 种群合并: 设定种群合并条件为当种群分裂过

程中任一目标的历史最优值发生更新, 或达到最大分裂代数 $\eta_s (\eta_s = \eta_g = \eta)$, 合并 SP1 与 SP2 并更新 Pareto 档案集。种群在合并状态下将恢复进行全局搜索。

2.5 算法流程

为保持种群多样性和分散性, 算法在初始化时工序、机器人、助理子序列均为随机生成。此外, 为避免概率参数的过饱和, 每次合并种群时对概率参数进行重置。算法流程如图 9 所示。

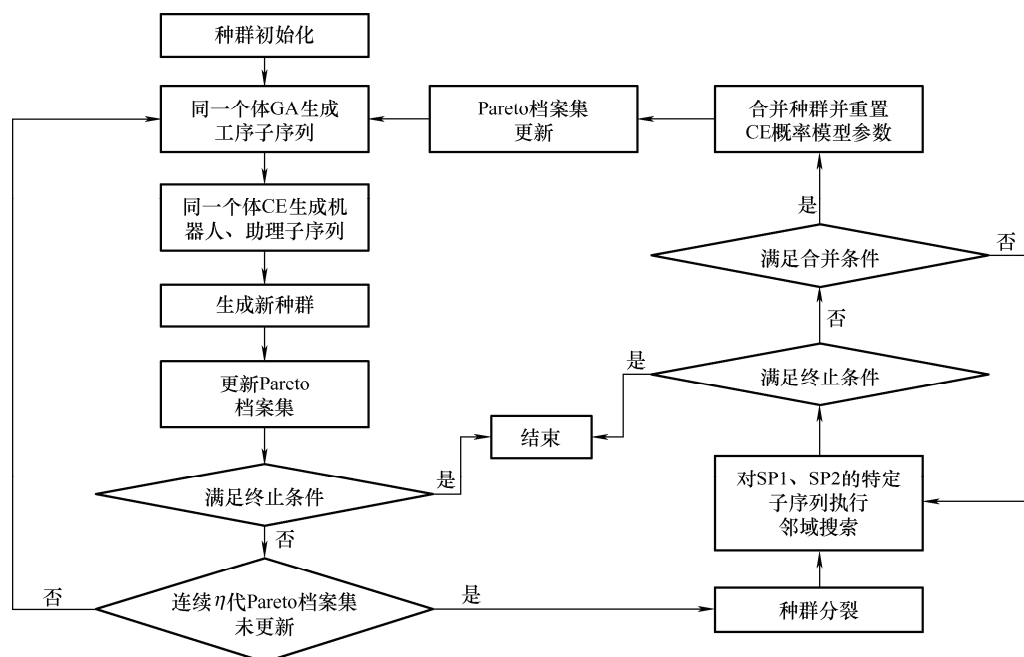


图9 算法流程图

3 试验设计与结果分析

3.1 相关资源数据设置

采用若干规模的 U 形装配线平衡标准案例(来自专业数据支持网站: <http://alb.mansci.de/>)对 RCUALBP_MRC 问题进行相关试验和分析。对于问题中相关资源的数据, 包括机器人, 助理和普通工人, 参照 JAYASWAL 等^[19]提出的方法, 以及本文模型求解实际情况的情况, 设置如下。

$$t_{ir} = (100 - \varepsilon)\% \times t_{i0}, \forall r \in R \quad (29)$$

$$t_{ia} = (100 - \theta)\% \times t_{i0}, \theta \sim U(20, 40) \quad (30)$$

$$CJ = 100, CW = 30, CA = 45 \quad (31)$$

不同类型的机器人对不同工序操作时间改善的效率 ε 都不同, 表 2 列举了不同类型机器人对应的 ε 区间和成本, 式(29)是机器人操作时间的计算; 式(30)是助理协助下操作时间的计算, 改善效率 θ 同样在区间内随机浮动; 式(31)给出了工位, 普通工人和助理的成本。表 3 是不同问题规模下机器人和助理的

数量约束。表 4 给出了标准测试案例 WARNECKE_58 部分工序的相关数据, ε 和 θ 均为各自区间内随机生成。

表2 不同机器人类型对应的效率和成本

机器人类型	效率 ε	成本
R_1	$\varepsilon \sim U(10, 30)$	30
R_2	$\varepsilon \sim U(20, 40)$	45
R_3	$\varepsilon \sim U(30, 50)$	60

表3 不同问题规模下的机器人-助理数量约束

I	NR_1	NR_2	NR_3	NR	NA
58	5	4	3	12	6
75	6	4	4	14	7
83	6	4	4	14	7
89	6	4	4	14	7
94	6	4	4	14	7
111	6	5	5	16	8

3.2 算法参数设置

算法包含四个关键参数: 统计样本的 ρ 分位数, 机器人与助理的概率参数的平滑系数 α 和 β , 种群

最大分裂代数 η , 如表 5 所示。采用 DOE 的方法^[35]分析参数对于算法性能的影响。

表 4 标准案例 WARNECKE_58 的部分数据设置

i	优先约束工序	t_{i0}	ε_1	ε_2	ε_3	θ
1	—	10	14	20	37	29
10	4	52	16	36	39	35
15	11	12	27	30	35	37
25	2, 20	24	24	34	48	29
30	27	22	14	21	34	33
40	37	7	26	26	33	35
45	32, 43, 44	43	12	27	34	37
55	48, 54	33	16	20	30	27

表 5 参数水平表

参数	水平			
	1	2	3	4
ρ	10	20	30	40
α	0.1	0.2	0.3	0.4
β	0.1	0.2	0.3	0.4
η	5	10	15	20

各参数均设置四个水平, 如表 6 所示, 建立规模为 $L_{16}(4^4)$ 的正交试验表。算法由 Delphi2010 编程实现, 试验环境为 Windows10 操作系统, Intel Core i5-8500 @4.10 GHz 处理器, 8 GB 内存。选取中等规模算例 ARC1_83, 节拍时间设置为 3 786, 种群规模设定为 100, 各参数组合下独立运行算法 20 次, 每次运行的时间为 $1.5 \times n$ (n 为算例的规模, 单位 s)。

为了更为客观地反映各参数组合对算法性能的影响, 采用两个指标的加权值作为该参数组合的最终响应值(Response value, RV)。第一个指标衡量的是算法获得非支配解的能力: 每组参数下算法每次运行产生一组非支配解集, 运行 20 次后将所有非支配解集合并得到一个非支配解集的集合, 将该集合中不被其余解支配的解的个数除以集合中解的总数, 获得的百分比即第一个响应指标 RV1; 第二个指标衡量的是算法获得的非支配解的质量, 每组参数组合下独立运行算法 20 次后得到的所有不被支配的解构成该参数组合下最终的非支配解集, 与其余所有参数组合下的非支配解集进行比较, 采用 ISHIBUCHI 等^[34]提出的评价方法, 描述如下

$$R_N(S_j) = \left| S_j - \left\{ x \in S_j \mid \exists y \in S : y \prec x \right\} \right| / |S_j| \quad (32)$$

其中 S 表示 k 组非支配解集的集合, $S = S_1 \cup \dots \cup S_k$, $y \prec x$ 表示 x 被 y 支配, $|S_j|$ 表示非支配解集 S_j 中非支配解的个数, R_N 称为占优比, 表示解集 S_j 中

不被其他解集支配的解所占的比例。每组参数下的 R_N 即为第二个响应指标 RV2, 将 RV1 与 RV2 按式(33)进行加权得到各组参数组合的总响应值 RV

$$RV = 0.5 \times (RV1 + RV2) \quad (33)$$

表 6 参数组合正交试验表

组合	参数水平				RV
	ρ	α	β	η	
1	1	1	1	1	4.41
2	1	2	2	2	19.44
3	1	3	3	3	3.45
4	1	4	4	4	9.26
5	2	1	2	3	3.45
6	2	2	1	4	3.45
7	2	3	4	1	8.62
8	2	4	3	2	4.84
9	3	1	3	4	4.69
10	3	2	4	3	5.36
11	3	3	1	2	29.00
12	3	4	2	1	21.35
13	4	1	4	2	7.81
14	4	2	3	1	51.39
15	4	3	2	4	1.52
16	4	4	1	3	3.03

通过分析表 7 各参数组合的平均响应值以及图 10 各参数组合对算法性能的影响趋势, 不难看出参数 η 对算法性能的影响程度最大, 原因是适当程度的种群分裂与合并有利于算法平衡全局搜索与局部搜索; 其次是机器人分配概率参数的平滑系数 α , 由于 CE 的性能很大程度上取决于概率参数的收敛程度, 取值过大或过小都容易导致算法的早熟收敛或收敛过缓; ρ 分位数的取值大小影响了算法对于优质解信息的统计; 由于助理分配的取值范围较小(非 0 即 1)所以平滑系数 β 对于算法性能的影响较小。根据以上分析算法推荐的参数设置如下: $\rho=40$, $\alpha=0.2$, $\beta=0.3$, $\eta=5$ 。

表 7 各参数组合的平均响应值和等级

水平	ρ	α	β	η
1	9.14	5.09	9.97	21.44
2	5.09	19.91	11.44	15.27
3	15.10	10.65	16.09	3.82
4	15.94	9.62	7.76	4.73
极差	10.85	14.82	8.33	17.62
等级	3	2	4	1

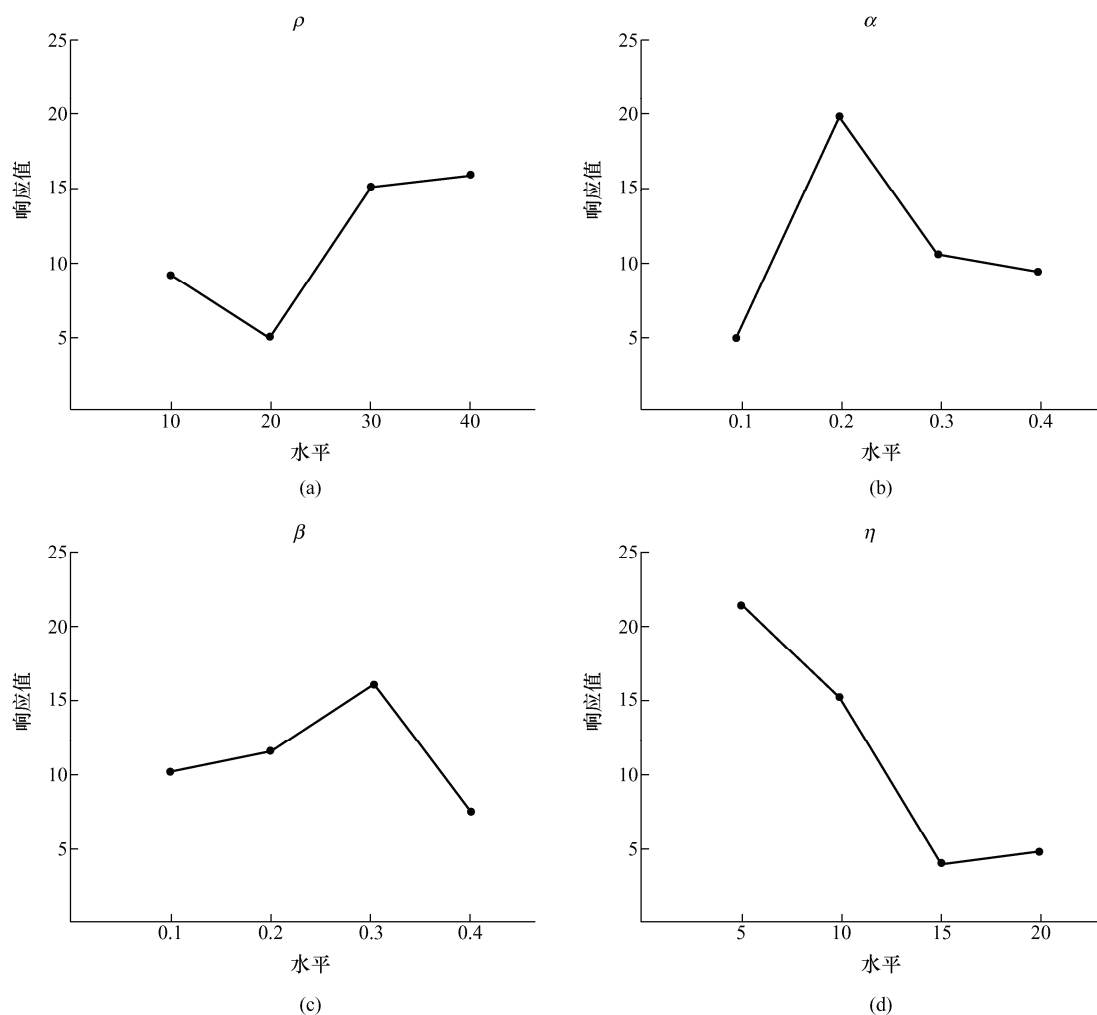


图10 各参数对算法性能的影响趋势

3.3 算法性能对比

算法性能对比试验将分为两个阶段进行, 第一阶段验证本文所采用 TSFBC 编码与协同进化操作的有效性。为了排除局部搜索对算法性能的影响, 更加直观的反映对比编码方式与全局搜索操作之间的性能差异, 以下第一阶段所有比较算法均去除了各自的局部搜索操作: ① 算法全局采用 CE-GA 协同搜索, 工序分配子序列采用基于工序选择因子编码的 CE-GACEA_TSFBC; ② 算法全局采用 CE-GA 协同搜索, 工序分配子序列采用基于多规则编码的 CE-GACEA_MRBC; ③ 算法全局采用遗传算法, 工序分配子序列采用 TSFBC 编码的 GA_TSFBC; ④ 算法全局采用 CE 方法, 工序分配子序列采用 TSFBC 编码的 CE_TSFBC。多目标问题以两项指标对算法性能进行综合评价, 第一项指标为式(32)中的非支配解占优比, 评价的是算法获得非支配解的质量。第二项指标为非支配解占优个数, 描述为非支配解集 S_j 中不被 S 中的任意解所支配的数量

$$N_N(S_j) = |S_j - \{x \in S_j | \exists y \in S: y \prec x\}| \quad (34)$$

显然, $R_N(S_j)$ 与 $N_N(S_j)$ 的值越大说明算法 j 求得的非支配解集的表现越好, 算法性能越优秀。仿真试验算法参数设置与运行环境参照第 3.2 节, 选取规模 n 为 58~111 的不同算例, 对每个算例设置 4 组节拍时间, 种群规模设置为 100, 所有对比算法在相同运行时间 $1.5 \times n$ (单位: s) 下各独立运行 20 次, 取 R_N 和 N_N 的平均值, 第一阶段算法性能对比结果如表 8 所示。

分析表 8 所示的试验结果, 在算法全局同样采用 CE-GA 协同搜索时, 工序分配子序列采用 TSFBC 编码的 CE-GACEA_TSFBC 性能明显优于采用 MRBC 编码的 CE-GACEA_MRBC。相较于对工序分配规则建立排序模型, 解码过程涉及对应规则计算的 MRBC 编码, TSFBC 编码直接对工序序号建立排序模型, 解码时按选择因子大小选择并分配工序, 无需复杂的计算, 很大程度上提升了编码解码的效率。此外, 在使用相同编码的条件下对比算法全局, 采用 CE-GA 协同搜索的 CE-GACEA_TSFBC 在绝大多数规模算例下的性能也优于采用单一遗传

遗传算法或 CE 方法生成全部子序列的 GA_TSFBC 以及 CE_TSFBC。这表明采用 GA 和 CE 协同进化可丰富搜索, 有利于发现优质解区域, 从而能获得更好的解。

算法性能对比试验的第二阶段将融入双种群协同局部搜索操作的 CE-GA_CEA 与近年来重要国际期刊中的多目标优化算法进行对比。IMMOGLS^[34]是一种基于随机权的多目标遗传算法, 通过多目标的随机加权作为适应值选择优势个体, 其性能被证明优于著名的多目标遗传算法 NSGA-II 和 SPEA。ITLBO 是一种改进的离散教与学优化算法^[36], 个体通过教学和互学阶段向优势个体学习, 并在自学阶段融合延迟爬山算法的机制对自身进行局部搜索。

BNSGA-II^[16]基于 Benders 分解法构造可行解, 并通过基于概率的层次化遗传操作扩充邻域结构增强算法的寻优能力, 已被验证是求解多目标 UALBP-1 的一类有效算法。IMMOGLS 与 ITLBO 此前尚未应用于 UALBP 的求解, 因此试验中这两种算法的工序子序列将统一采用 TSFBC 编码, BNSGA-II 则保留原文所采用的 MRBC 编码。由于对比算法均带有局部搜索操作, 增加算法单次运行时间至 $3 \times n$ (n 为算例的规模, 单位 s), 其余参数设置与第一阶段试验相同, 分别独立运行 20 次取 R_N 和 N_N 的平均值。试验结果如表 9 所示, CE-GA_CEA 对不同规模问题求解的性能都优于其他对比算法, 验证了算法在求解 RCUALBP_MRC 的有效性。

表 8 第一阶段算法性能对比试验结果

n	CT	CE-GACEA_TSFBC		CE-GACEA_MRBC		GA_TSFBC		CE_TSFBC	
		R_N	N_N	R_N	N_N	R_N	N_N	R_N	N_N
58	54	0.925 0	3.30	0	0	0.200 0	0.25	0	0
58	65	0.975 0	1.55	0	0	0.325 0	0.55	0	0
58	82	0.970 8	2.30	0	0	0.400 0	0.55	0.051 7	0.20
58	111	0.800 0	1.25	0	0	0.833 3	1.20	0.066 7	0.20
75	40	0.950 0	1.40	0	0	0.266 7	0.40	0	0
75	45	0.975 0	2.90	0	0	0.350 0	0.45	0.010 0	0.05
75	50	0.958 3	2.05	0	0	0.500 0	0.80	0	0
75	56	0.975 0	1.70	0.016 7	0.05	0.158 3	0.30	0	0
83	3786	0.937 5	1.70	0	0	0.266 7	0.45	0.016 7	0.05
83	5048	0.908 3	1.50	0	0	0.537 5	0.85	0.026 7	0.10
83	6842	0.675 0	1.00	0	0	0.770 8	1.25	0.235 0	0.75
83	8412	0.537 5	1.00	0.012 5	0.05	0.700 0	0.85	0.248 6	0.95
89	75	0.971 4	1.90	0	0	0.175 0	0.20	0	0
89	87	0.920 0	2.10	0	0	0.100 0	0.20	0.016 7	0.05
89	103	0.920 8	1.75	0	0	0.408 3	0.65	0.041 7	0.10
89	127	0.887 5	1.80	0	0	0.300 0	0.40	0.050 0	0.15
94	176	0.975 0	2.50	0	0	0.175 0	0.35	0	0
94	211	1.000 0	1.95	0	0	0.100 0	0.10	0	0
94	263	0.975 0	1.50	0	0	0.391 7	0.55	0	0
94	351	0.845 0	1.20	0	0	0.500 0	0.65	0.107 5	0.30
111	5755	0.806 1	3.35	0	0	0.500 0	0.90	0.025 0	0.05
111	6837	1.000 0	1.95	0	0	0.150 0	0.20	0	0
111	8847	1.000 0	1.55	0	0	0.295 8	0.40	0.075 0	0.10
111	11570	0.962 5	1.50	0	0	0.175 0	0.30	0.062 5	0.15

4 结论

(1) 建立了 RCUALBP_MRC 的问题模型。

(2) 设计基于概率模型的交叉熵方法, 可有效学习优质解的模式信息并指导搜索。

(3) 构建基于交叉熵方法与遗传算法的协同进

化框架, 可丰富算法的搜索行为, 有利于搜索解空间中更多的不同区域, 从而能在较短时间内找到问题的优质解或近似最优解。

(4) 在算法的局部搜索阶段加入种群分裂-合并机制, 分裂可对解空间特定区域执行更深入的搜索, 合并可确保子种群对不同搜索区域的较均匀覆盖, 从而有效地平衡算法的全局与局部

表 9 第二阶段算法性能对比试验结果

n	CT	CE-GACEA		IMMOGLS		ITLBO		BNSGA-II	
		R_N	N_N	R_N	N_N	R_N	N_N	R_N	N_N
58	54	1.000 0	1.85	0	0	0	0	0	0
58	65	0.950 0	1.45	0.150 0	0.35	0	0	0	0
58	82	0.975 0	1.85	0.150 0	0.25	0.050 0	0.15	0	0
58	111	0.825 0	1.30	0.300 0	0.40	0.205 0	0.55	0	0
75	40	1.000 0	1.65	0.100 0	0.15	0	0	0	0
75	45	1.000 0	1.75	0.175 0	0.25	0	0	0	0
75	50	0.883 3	1.70	0.225 0	0.30	0	0	0	0
75	56	0.950 0	1.45	0.183 3	0.35	0	0	0	0
83	3786	1.000 0	1.40	0.266 7	0.35	0	0	0	0
83	5048	0.925 0	1.75	0.450 0	0.70	0.170 0	0.40	0	0
83	6842	0.875 0	1.40	0.400 0	0.60	0.251 8	0.75	0	0
83	8412	0.900 0	1.30	0.283 3	0.45	0.253 3	0.65	0	0
89	75	1.000 0	1.55	0.150 0	0.15	0.012 5	0.05	0	0
89	87	0.891 7	1.65	0.283 3	0.55	0.087 5	0.15	0	0
89	103	0.950 0	1.20	0.433 3	0.75	0.093 7	0.35	0	0
89	127	0.925 0	1.60	0.175 0	0.25	0.144 6	0.50	0	0

搜索。

未来工作是针对不确定多资源约束 U 形装配线平衡问题, 研究有效的协同进化机制, 并设计高效的 CE-GACEA 进行求解。

参 考 文 献

- [1] 张则强, 程文明, 钟斌, 等. 混合品种装配线平衡问题的一种混合搜索机制的蚁群算法[J]. 机械工程学报, 2009, 45(5): 95-101.
ZHANG Zeqiang, CHENG Wenming, ZHONG Bin, et al. Hybrid behavior ant colony optimization for mixed-model assembly line balancing problem [J]. Journal of Mechanical Engineering, 2009, 45(5): 95-101.
- [2] SALVESON M E. The assembly line balancing problem[J]. Journal of Industrial Engineering, 1955, 29(10): 55-101.
- [3] SCHOLL A. Balancing and sequencing of assembly lines[M]. 2nd ed. Heidelberg: Physica-Verlag, 1999.
- [4] MILTENBURG G J, WIJNGAARD J. The U-line balancing problem[M]. Maryland: INFORMS, 1994.
- [5] MILTENBURG J. U-shaped production lines: A review of theory and practice[J]. International Journal of Production Economics, 2001, 70(3): 201-214.
- [6] 胡中扬. 走进丰田—精益生产的缔造者[EB/OL]. [2018-6-26]. <http://articles.eworks.net.cn/plmoverview/article141623.htm>.
- [7] MUKUND N J, PONNAMBALAM S G. Robotic u-shaped assembly line balancing using particle swarm optimization[J]. Engineering Optimization, 2015, 48(2): 1-22.
- [8] LIU C M, CHEN C H. Multi-section electronic assembly line balancing problems: A case study[J]. Production Planning & Control, 2002, 13(5): 451-461.
- [9] URBAN T L. Optimal Balancing of U-Shaped Assembly Lines[J]. Management Science, 1998, 44(5): 738-741.
- [10] GÖKÇEN H, AĞPAK K, GENCER C, et al. A shortest route formulation of simple U-type assembly line balancing problem[J]. Applied Mathematical Modelling, 2005, 29(4): 373-380.
- [11] 唐秋华, 林斌, 何晓霞, 等. 基于随机机会约束规划的 U 形装配线平衡优化[J]. 计算机集成制造系统, 2016, 22(4): 955-964.
TANG Qiuhua, LIN Bin, HE Xiaoxia, et al. Balancing optimization of U-shaped assembly lines based on stochastic chance constraint programming[J]. Computer Integrated Manufacturing Systems, 2016, 22(4): 955-964.
- [12] WAINWRIGHT D A A R L. Applying genetic algorithms to the U-shaped assembly line balancing problem[C] //IEEE World Congress on Computational Intelligence, May 4-9, 1998, Anchorage. IEEE, 1998: 96-101.
- [13] BAYKASOĞLU A, OZBAKIR L. Stochastic U-line balancing using genetic algorithms[J]. International Journal of Advanced Manufacturing Technology, 2007, 32(1-2): 139-147.
- [14] RABBANI M, KAZEMI S M, MANAVIZADEH N.

- Mixed model U-line balancing type-1 problem: A new approach[J]. *Journal of Manufacturing Systems*, 2012, 31(2): 131-138.
- [15] ZHANG Zikai, TANG Qiuhua, HAN Dayong, et al. Enhanced migrating birds optimization algorithm for U-shaped assembly line balancing problems with workers assignment[J]. *Neural Computing and Applications*, 2019, 31(11): 7501-7515.
- [16] 张子凯, 唐秋华, 张利平. 多重资源约束下的多目标 U 形装配线平衡[J]. *计算机集成制造系统*, 2017, 23(11): 100-110.
- 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.
- [17] OKSUZ M K, BUYUKOZKAN K, SATOGLU S I. U-shaped assembly line worker assignment and balancing problem: A mathematical model and two meta-heuristics[J]. *Computers & Industrial Engineering*, 2017, 112: 246-263.
- [18] KARA Y, OZGUVEN C, YALSIN N, et al. Balancing straight and U-shaped assembly lines with resource dependent task times[J]. *International Journal of Production Research*, 2011, 49(21): 6387-6405.
- [19] JAYASWAL S, AGARWAL P. Balancing U-shaped assembly lines with resource dependent task times: A simulated annealing approach[J]. *Journal of Manufacturing Systems*, 2014, 33(4): 522-534.
- [20] 周济. 智能制造——“中国制造 2025”的主攻方向[J]. *中国机械工程*, 2015, 26(17): 2273-2284.
- ZHOU Ji. Intelligent manufacturing——main direction of “Made in China 2025”[J]. *China Mechanical Engineering*, 2015, 26(11): 2273-2284.
- [21] RUBINSTEIN R Y. Optimization of computer simulation models with rare events[J]. *European Journal of Operational Research*, 1997, 99(1): 89-112.
- [22] CASERTA M, RICO E Q. A cross entropy algorithm for the Knapsack problem with setups[J]. *Computers & Operations Research*, 2008, 35(1): 241-252.
- [23] CHEPURI K, HOMEM-DE-MELLO T. Solving the vehicle routing problem with stochastic demands using the cross-entropy method[J]. *Annals of Operations Research*, 2005, 134(1): 153-181.
- [24] WANG Guirong, LI Qiqiang, WANG Luhao. An improved cross entropy algorithm for steelmaking-continuous casting production scheduling with complicated technological routes[J]. *Journal of Central South University*, 2015, 22(8): 2998-3007.
- [25] J.EDRZEJOWICZA P, SKAKOVSKI A. A cross-entropy-based population-learning algorithm for discrete-continuous scheduling with continuous resource discretization[J]. *Neurocomputing*, 2010, 73(4-6): 655-660.
- [26] BUDI S, MUHAMMAD A B, STEFANUS E W. A cross entropy-genetic algorithm form-machines no-wait job-shop scheduling problem [J]. *Journal of Intelligent Learning Systems and Applications*, 2011, 3: 171-180.
- [27] 李国成, 肖庆宪. 求解高维函数优化问题的交叉熵蝙蝠算法[J]. *计算机工程*, 2014, 40(10): 168-174.
- LI Guocheng, XIAO Qingxian. Cross-entropy bat algorithm for solving high-dimensional function optimization problem[J]. *Computer Engineering*, 2014, 40(10): 168-174.
- [28] 王凌, 沈婧楠, 王圣尧, 等. 协同进化算法研究进展[J]. *控制与决策*, 2015, 30(2): 193-202.
- WANG Ling, SHENG Jingnan, WANG Shengyao, et al. Advances in co-evolutionary algorithms[J]. *Control and Decision*, 2015, 30(2): 193-202.
- [29] LEI D. Co-evolutionary genetic algorithm for fuzzy flexible job shop scheduling[J]. *Applied Soft Computing*, 2012, 12(8): 2237-2245.
- [30] WANG Ling, WANG Shengyao, XU Ye, et al. A bi-population based estimation of distribution algorithm for the flexible job-shop scheduling problem[J]. *Computers & Industrial Engineering*, 2012, 62: 917-926.
- [31] 刘冉, 楼佩煌, 唐敦兵, 等. 解决 U 形装配线平衡调度问题的免疫协同进化算法[J]. *中国机械工程*, 2010, 21(7): 815-821.
- Liu Ran, LOU Peihuang, TANG Dunbing, et al. An immune coevolution algorithm for balancing and sequencing on mixed-model U-lines [J]. *China Mechanical Engineering*, 2010, 21(7): 815-821.
- [32] KIM Y K, KIM J Y, KIM Y. A coevolutionary algorithm for balancing and sequencing in mixed model assembly lines[J]. *Applied Intelligence: The International Journal of Artificial, Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, 2000, 13(3): 247-258.
- [33] DEB K, PRATAP A, AGARWAL S, et al. A fast and elitist multi objective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 0-197.
- [34] ISHIBUCHI H. Balance between genetic search and local

search in memetic algorithms for multiobjective permutation flowshop scheduling[J]. IEEE Trans. Evol. Comput., 2003, 7(2): 204-223.

[35] MONTGOMERY D C. Design and analysis of experiments[M]. Hoboken: John Wiley and Sons, 2005.

[36] LI Dashuang, ZHANG Caoyong, SHAO Xinyu, et al. A multi-objective TLBO algorithm for balancing two-sided

assembly line with multiple constraints[J]. Journal of Intelligent Manufacturing, 2016, 27(4): 725-739.

作者简介: 郑逸凡, 男, 1994 年出生。主要研究方向为智能算法与优化调度。

E-mail: zyf_evan777@163.com

钱斌(通信作者), 男, 1976 年出生, 教授, 博士研究生导师。主要研究方向为优化调度理论与方法。

E-mail: bin.qian@vip.163.com

第 9 届上银优秀机械博士论文奖——佳作奖

数控机床多轴联动铣削加工动力学集成建模研究

作 者: 张 星

毕业学校: 西安交通大学

指导教师: 赵万华

高效精密数控加工与机床进给系统运动和切削过程密切相关, 传统上针对数控机床和切削工艺的研究往往都是相对独立开展的, 致使未能完全揭示数控加工的本质。本论文重点研究数控机床多轴联动铣削加工动力学集成建模方法, 对揭示多轴联动、铣削加工与零件加工精度和效率之间的量化关系与影响机制, 理论上突破机床-工艺交互作用机理, 应用上实现高精高效加工具有重要意义。

建立了复杂曲面多轴加工过程铣削力精确预测模型。综合考虑刀具偏心、刀具姿态和联动加工进给速度等因素, 实现了刀具-工件啮合区瞬时切削层厚度的准确求解。提出了基于电涡流传感器测量的刀具偏心非接触辨识方法, 发现了刀具偏心具有主轴转速依赖特性。提出了基于薄板铣削试验的刀具-工件材料对比切力系数的三步分离标定方法。实现了立铣刀、球头刀等多种刀具在复杂工况条件下的铣削力精确预测。

提出了柔性刀具-工件系统变姿态加工多阶铣削动力学建模与分析一般性方法。建立了多轴加工过程中任意刀具-工件相对姿态下的再生切削层厚度通用计算模型, 实现了考虑系统多阶模态特性、动力学模型自由度方向与刀具-工件啮合坐标系非同向特性、以及刀具和工件系统主导振动非同向特性的铣削动力学行为表征。分析了系统模态特性、刀具姿态、偏心状态、铣削参数等对柔性加工系统自激振动和强迫振动力学行为的影响规律。

研究了数控机床多轴联动加工进给运动与铣削过程的动力学集成建模方法。建立了数控机床多轴联动的运动控制模型, 实现了多轴联动加工过程中进给系统非理想位移输出条件下的刀具-工件动态啮合过程表征, 形成了复杂曲面铣削成形过程的物理仿真方法。揭示了集成系统在数控指令、电机谐波力矩、动态铣削力等多源激励作用下零件已加工表面轮廓高、中、低频特性的形成机制、耦合关系及演变规律。

实现了考虑机床联动加工性能的工艺过程铣削参数与刀具路径优化。构建了以材料去除率最大化为优化目标, 包含参数可行域、铣削稳定性、加工精度、切削扭矩和主轴功率等多重约束条件的数值优化问题, 实现了无颤振高效高精加工参数获取。针对复杂曲面轮廓特征, 以加工误差最小化为目标, 提出了基于控制点修正的刀具路径全局优化方法, 实现了高精度要求下的刀具路径优化。