

文章编号: 1006-3080(2025)05-0633-12

DOI: 10.14135/j.cnki.1006-3080.20241225001

多工位、多机器人焊接任务分配与路径规划

王 烨, 王学武, 顾幸生

(华东理工大学能源化工过程智能制造教育部重点实验室, 上海 200237)

摘要: 针对实际汽车制造业中多工位、多机器人生产线的焊接任务分配、焊接顺序规划, 以及工件和生产线组成的众多约束条件, 本文建立了多工位、多机器人焊接任务分配与路径规划 (Multi-Station Multi-Robot Welding Task Assignment and Path Planning, MSMR-WTAPP) 数学模型; 优化目标是同时最小化生产线第一个工件加工时间、后续工件加工时间和机器人运动路径长度; 提出了一种基于个体种群密度 (Individual Population Density, IPD) 的改进 SPEA2 算法 (SPEA2+IPD); 针对焊接任务分配和焊接顺序规划的耦合问题设计了双层编码方案, 并研究了机器人工序的解码过程。通过仿真实验验证了 SPEA2+IPD 算法在优化多工位、多机器人生产线节拍、效率和焊接路径等方面的有效性和优越性。SPEA2+IPD 算法优化得到的生产节拍与工厂实际生产节拍相比, 第一个工件时间缩短了 20.7%, 后续每个工件的生产时间都减少 15.2%, 说明提出的模型和算法对优化工厂生产具有实际意义。

关键词: 汽车制造; 工业机器人; 焊接; 任务分配; 路径规划

中图分类号: TP241.2

文献标志码: A

工业机器人的出现引发了各行各业的革命, 激发了人们对制造业自动化的极大兴趣。机器人通过自动化焊接和高效操作大大提高了制造业的生产效率, 对制造业的发展起着至关重要的推动作用^[1-2]。机器人路径规划是单工位机器人自动化生产的主要问题, 需要确定机器人的焊接顺序和路径以实现期望的焊接结果, 已经有许多学者提出了有效的机器人路径规划方法。汤彬等^[3] 改进了蚁群算法和粒子群算法, 解决了白车身双焊接机器人的避障路径规划问题。王学武等^[4] 改进了分区粒子群算法, 在车门生产中对点焊机器人进行路径规划, 得到了焊接的最短路径。Jiang 等^[5] 提出了一种改进的 RRT 方法与人工势场方法结合, 采用自适应引力场、动态步长和新的节点约束策略, 有效规划出了综合性能最佳的安全最优避障路径。

在典型的机器人装配焊接生产线中, 多个工位有序地完成所有生产阶段。每个工位的作业空间通常被设计为由一个或多个机器人共享, 这些机器人

可以在工位上同时协作完成复杂的任务^[6-7]。对于多工位多机器人生产线 (Multi-station multi-robot production line), 生产方式是批量生产同型号产品。在这种模式下, 每个产品的生产过程都是一致的, 并且每个产品的加工时间也是一致的。但是一个产品最早的开始时间可以在前一个产品完成加工之前的某个时间点, 从第一个工件加工完成后, 每个工件都在间隔相同的时间后被加工完成。制造商的目的是最大化生产率, 从而最小化一批产品的总生产周期。因此, 优化生产线第一个产品开始加工的时间和后续加工每个产品需要的间隔时间, 对于缩短产品的整个生产周期至关重要。

工业中自动化生产模式显著提高了生产率, 机器人的应用越来越频繁, 同时也在工位和机器人之间的分配、协调和调度方面提出了挑战^[8]。该领域的研究从单个机器人到多个机器人, 再到对多个工位和多个机器人生产线进行研究^[9]。多工位和机器人生产线的优化存在多种分配和调度问题, 包括机器

收稿日期: 2024-12-25

基金项目: 国家自然科学基金 (61973120, 62076095)

作者简介: 王 烨 (1995—), 男, 山西长治人, 博士生, 主要研究方向为工业机器人生产线优化、多目标优化。E-mail: yewang@mail.ecust.edu.cn

通信联系人: 顾幸生, E-mail: xsgu@ecust.edu.cn

引用本文: 王 烨, 王学武, 顾幸生. 多工位、多机器人焊接任务分配与路径规划 [J]. 华东理工大学学报(自然科学版), 2025, 51(5): 633-644.

Citation: WANG Ye, WANG Xuewu, GU Xingsheng. Multi-Station Multi-Robot Welding Task Assignment and Path Planning[J]. Journal of East China University of Science and Technology, 2025, 51(5): 633-644.

人和工位的任务分配、任务序列规划、机器人路径规划和工序过程调度等^[10-11]。国内外一些学者对单个机器人或者一个工位拥有几个机器人的简单子问题耦合进行了研究。Touzani 等^[12]针对多机器人任务排序问题,提出了一种快速生成有序近最优解的算法。将问题建模为一个最小和最大多重广义旅行商问题,最大限度地减少了生产时间以及机器人的整体运动持续时间。Zhao 等^[13]研究了汽车装配线多机器人协同完成点焊任务,保证相邻机器人被分配不同的焊点集合,减少了每个机器人完成焊接任务的时间。

在文献中大多只涉及有限的工业场景,只能解决部分生产线的优化问题,然而实际的生产线需要考虑多个耦合问题的优化。比如:生产线组成有多个工位,需要将焊点和焊缝合理地分配在各个工位;单个工位有两个及以上可达机器人,需要为该工位分配机器人,同时为机器人分配焊接任务并规划焊接顺序;生产线有多个机器人,需要解决机器人的工序调度,使它们不产生冲突。

实际生产线多个耦合子问题通常需要考虑多个优化的目标,这些目标往往互相冲突并且需要同时优化。随着问题的规模和耦合性的增加,优化问题与生产线约束之间的相互作用更加复杂,比如工位夹具、工件装配、焊接任务可达性、机器人工序时间约束等相关要求。因此,该问题的有效优化不仅在于计算,还需要创新的思维分解问题与约束来实现优化的可行性。关注复杂生产线中多个子问题的耦合,对多个目标同时优化找到一组最优的 Pareto 解集,为工厂提供更多的可选择生产方案,具有重要的现实意义。本文针对真实的工业场景,考虑了不同子问题耦合应用场景的约束条件,设计了双层编码方案分解耦合问题,提出了 SPEA2+IPD 算法对问题

进行优化,解决了任务分配、机器人焊接序列规划、机器人工序调度等子问题互相耦合的多工位、多机器人焊接任务分配与路径规划(MSMR-WTAPP)问题。

1 MSMR-WTAPP

1.1 生产线和工件描述

本文研究了一个汽车工业中的机器人生产线,这个生产线的 Catia 模型如图 1 所示,主要由 3 个装配工位(J_1 、 J_2 和 J_3)、一个转运工位(J_4)、一个卸料工位(J_5)、3 个焊机工位(G_1 、 G_2 和 G_3)和 4 个机器人(R_1 、 R_2 、 R_3 和 R_4)组成。机器人 R_1 和 R_2 既可以焊接,也可以在两个工位之间搬运工件。机器人 R_3 和 R_4 抓取工件在焊机工位焊接。零件在装配工位人工组装后由机器人焊接,3 个装配工位和焊机工位 G_1 、 G_2 都是点焊。焊机工位 G_3 是螺柱焊工位,由 3 种螺柱焊枪(S_1 、 S_2 和 S_3)组成。

该生产线设计为大批量制造图 2 所示的汽车门槛组成。每个工件的制造过程都是相同的生产工序,生产过程需要多个零件组装,并依次经过 J_1 、 J_2 、 J_3 、 G_1 、 G_2 、 G_3 6 个工位焊接,共包括 89 个焊点和 9 个螺柱。工件在 J_1 、 J_2 和 J_3 、 J_4 之间的移动分别由机器人 R_1 和 R_2 搬运。工位 J_3 完成的工件由机器人 R_3 抓起在工位 G_1 焊接并放在工位 J_4 ,然后机器人 R_4 从工位 J_4 抓起工件,在 G_2 和 G_3 焊接并放在工位 J_5 ,最后在 J_5 工位卸载工件。

该生产线机器人焊接一个焊点和一个螺柱的时间分别为 w_pT 和 w_sT ,机器人在两种工位的焊点或螺柱之间移动时间分别为 mTJ 和 mTG 。另外,工位组装零件时间为 ITJ_a ,机器人抓件时间为 luT ,放件时间为 pdT 。生产完成后,人工卸载时间为 ulT 。具体人工操作和机器人操作所需时间分别如表 1、表 2

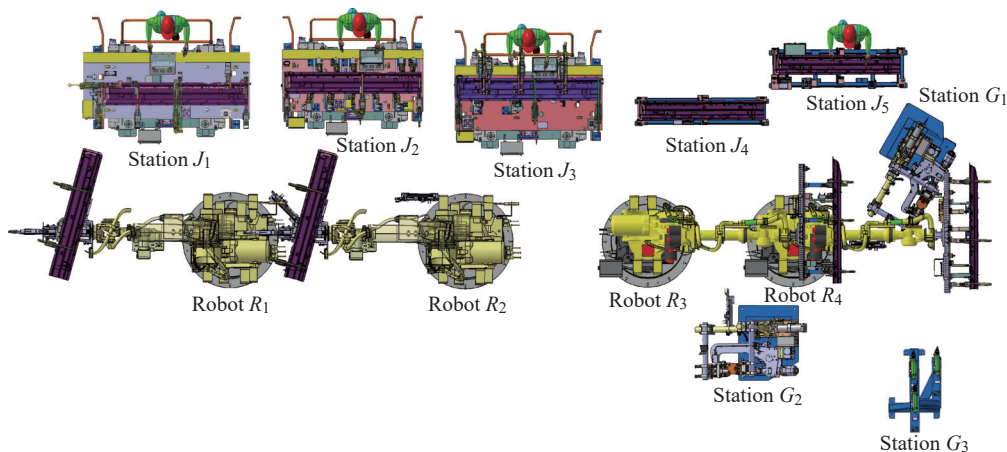


图 1 生产线 Catia 模型示意图

Fig. 1 Catia model representation of the production line

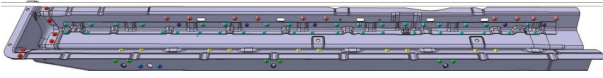


图 2 工件图

Fig. 2 Representation of workpiece

表 1 人工操作时间

Table 1 Manual operation time

Name	Time/s
ITJ ₁	25
ITJ ₂	30
ITJ ₃	20
ulT	6

表 2 机器人操作时间

Table 2 Robot operation time

Name	Time/s
luT or pdT	8
wpT or mTG	1.5
wsT	3
mTJ	2

表 3 任务分配无法焊接约束

Table 3 Unable to weld constraints for task assignment

Notation	Welding task set
UWJ ₁	$\{W_1, \dots, W_{10}, W_{14}, W_{15}, W_{18}, W_{20}, W_{21}, W_{24}, W_{29}, W_{31}, W_{41}, W_{48}, W_{49}, W_{52}, W_{59}, \dots, W_{88}\}$
UWJ ₂	$\{W_{15}, W_{18}, W_{20}, W_{22}, W_{29}, W_{31}, \dots, W_{34}, W_{38}, \dots, W_{46}, W_{49}, \dots, W_{59}, W_{70}, \dots, W_{88}\}$
UWJ ₃	$\{W_5, W_7, W_{11}, W_{13}, W_{14}, W_{19}, W_{25}, W_{26}, W_{36}, W_{37}, W_{40}, W_{43}, W_{44}, W_{47}, \dots, W_{61}, W_{70}, W_{71}, W_{73}, \dots, W_{75}, W_{77}, \dots, W_{79}, W_{81}, W_{82}, W_{85}, W_{86}, W_{88}\}$
UWG ₁	$\{W_3, W_4, W_{60}, \dots, W_{62}, W_{64}, W_{65}, W_{78}, W_{80}, W_{88}\}$
UWG ₂	$\{W_1, W_3, W_5, \dots, W_8, W_{16}, \dots, W_{18}, W_{21}, \dots, W_{23}, W_{27}, W_{30}, W_{35}, W_{38}, W_{40}, W_{43}, W_{45}, W_{55}, W_{56}, W_{72}, W_{73}, W_{76}, W_{77}, W_{80}, W_{81}, \dots, W_{84}, W_{87}, W_{89}\}$

表 4 任务分配强制焊接约束

Table 4 Enforce welding constraints for task assignment

Notation	Welding task set
EWJ ₁	$\{W_{12}, W_{18}, W_{42}, W_{51}, W_{58}\}$
EWJ ₂	$\{W_2, W_4, W_6, W_8, W_{10}, W_{60}, W_{62}, W_{64}, W_{66}, W_{68}\}$
EWJ ₃	$\{W_{72}, W_{76}, W_{80}, W_{87}\}$
EWS ₁	$\{W_{90}, W_{91}, \dots, W_{96}\}$
EWS ₂	$\{W_{97}\}$
EWS ₃	$\{W_{98}\}$

该 MSMR-WTAPP 问题的目标是找到各个工位和机器人最优的焊接任务分配和每个机器人的焊接

所示。

1.2 数学模型

生产过程可描述为各个工件一次经过工位 $J_a(a \in \{1, 2, 3\})$ 和 $G_b(b \in \{1, 2, 3\})$ 的焊接, 在工位 G_3 要经过 3 个螺柱焊枪 $S_c(c \in \{1, 2, 3\})$ 的焊接, 在每个工位焊接一部分焊点或者螺柱, 整个过程中所有焊点和螺柱都被焊接, 并且只焊接一次。各个工位的焊接任务和工件在工位之间的转移由机器人 $R_d(d \in \{1, 2, 3, 4\})$ 执行, 只要工件空闲或者机器人空闲就可以开始生产下一个工件, 所有工件都是相同的生产过程。工件的每个焊接任务表示为 $W_e(e \in \{1, 2, \dots, 98\})$, 焊点集合为 $SPW = \{W_1, W_2, \dots, W_{89}\}$, 螺柱集合为 $STW = \{W_{90}, W_{91}, \dots, W_{98}\}$ 。由于生产线的 3 个组装工位需要固定零件, 机器人的抓手也需要将零件固定, 每个工位都存在因为可达性导致无法焊接的焊点, 每个工位无法焊接的焊点集合表示为 UWJ_a 和 UWG_b , 如表 3 所示。工件在装配工位组装后必须焊接一部分焊点, 才可以保证零件不会在搬运中掉落, 不同类型的螺柱必须在对应螺柱焊枪焊接, 所以规定了各个位置必须焊接的焊点和螺柱集合 EWJ_a 和 EWS_c , 如表 4 所示。

顺序, 从而使第一个工件加工时间 (T_{first})、后续工件加工时间 ($T_{\text{subsequence}}$) 以及机器人路径长度 (PL) 最小。3 个目标函数的计算如式 (1)~(10) 所示, 式 (1)~(3) 为 T_{first} 的计算方法, $T_{\text{subsequence}}$ 根据式 (4) 和式 (5) 计算, 式 (6)~(10) 用来计算路径长度 PL。

$$F_1 = \min T_{\text{first}} = \min \left(\sum_{a=1}^3 (\text{ITJ}_a + \text{oTJ}_a) + \sum_{b=1}^3 \text{oTG}_b + 4 \times (\text{luT} + \text{pdT}) \right) + \text{ulT} \quad (1)$$

$$\text{oTJ}_a = \begin{cases} \text{sJ}_a N \times (\text{wpT} + \text{mTJ}) + \text{mTJ}, & a = 1, 3 \\ \max(\text{sJ}_a R_1 N, \text{sJ}_a R_2 N) \times (\text{wpT} + \text{mTJ}) + \text{mTJ}, & a = 2 \end{cases} \quad (2)$$

$$oTG_b = \begin{cases} sG_bN \times (wpT + mTG) + mTG_b, & b = 1, 2 \\ sG_bN \times (wsT + mTG), & b = 3 \end{cases} \quad (3)$$

oTJ_a 和 oTG_b 分别为机器人在 J_a 和 G_b 工位操作需要的时间, 操作时间包括焊接时间和移动时间, 是计算目标函数的中间变量, 由任务数量和单个任务焊接时间以及移动时间计算得到。

$$F_2 = \min T_{\text{subsequence}} = \min(\max(tR_1, tR_2, tR_3, tR_4)) \quad (4)$$

$$tR_d = \begin{cases} lTJ_d + oTJ_d + luT + \max(pdT + lTJ_{d+1} + oTJ_{d+1}, lTJ_d), & d = 1, 2 \\ luT + oTG_{d-2} + pdT, & d = 3 \\ luT + oTG_{d-2} + oTG_{d-1} + pdT, & d = 4 \end{cases} \quad (5)$$

决策变量 $J_aN = \{J_aN_1, J_aN_2, \dots, J_aN_x, \dots, J_aN_{sJ_aN}\}$ 和 $G_bN = \{G_bN_1, G_bN_2, \dots, G_bN_y, \dots, G_bN_{sG_bN}\}$ 分别表示分配到工位 J_a 和 G_b 的任务集合, 集合内元素的顺序为各工位的焊接顺序。 sJ_aN 和 sG_bN 分别表示集合 J_aN 和 G_bN 的大小。由于工位 J_2 为机器人 R_1 和 R_2 共同焊接, 决策变量 J_aR_dN 表示工位 J_2 的任务分配给机器人 R_d 的集合, 其中 $d \in \{1, 2\}$ 。 $J_aR_dN = \{J_aR_dN_1, J_aR_dN_2, \dots, J_aR_dN_z, \dots, J_aR_dN_{sJ_aR_dN}\}$, sJ_aR_dN 为集合 J_aR_dN 的大小, 机器人按照集合内元素的顺序焊接。

机器人必须在完成一个工件以后, 才可以开始加工下一个工件。完整的加工过程由 4 个机器人连续完成, 各个机器人加工的时间表示为 tR_d 。要使机器人没有时间冲突, 必须以 4 个阶段的最长时间作为生产节拍时间。生产线的一个工件结束后, 后续每个工件需要的额外加工时间为 $\max(tR_1, tR_2, tR_3, tR_4)$ 。

$$F_3 = \min PL = \min(\sum_{a=1}^3 (wPLJ_a + iniPLJ_a) + \sum_{b=1}^3 (wPLG_b + iniPLG_b)) \quad (6)$$

$$wPLJ_a = \begin{cases} \sum_{x=1}^{sJ_aN-1} \text{dis}(J_aN_x, J_aN_{x+1}), & a = 1, 3 \\ \sum_{d=1}^2 \sum_{z=1}^{J_aR_dN-1} \text{dis}(J_aR_dN_z, J_aR_dN_{z+1}), & a = 2 \end{cases} \quad (7)$$

$$wPLG_b = \sum_{y=1}^{sG_bN-1} \text{dis}(G_bN_y, G_bN_{y+1}), b = 1, 2, 3 \quad (8)$$

$$iniPLJ_a = \begin{cases} \text{dis}(iniR_1, J_aN_1) + \text{dis}(J_aN_{sJ_aN}, iniR_1), & a = 1 \\ \sum_{c=1}^2 \text{dis}(iniR_c, J_aR_dN_1) + \text{dis}(J_aR_dN_{sJ_aR_dN}, iniR_c), & a = 2 \\ \text{dis}(iniR_2, J_aN_1) + \text{dis}(J_aN_{sJ_aN}, iniR_2), & a = 3 \end{cases} \quad (9)$$

$$iniPLG_b = \begin{cases} \text{dis}(iniR_3, G_bN_1) + \text{dis}(G_bN_{sG_bN}, iniR_3), & b = 1 \\ \text{dis}(iniR_4, G_bN_1) + \text{dis}(G_bN_{sG_bN}, G_{b+1}N_1), & b = 2 \\ \text{dis}(G_bN_{sG_bN}, iniR_4), & b = 3 \end{cases} \quad (10)$$

路径长度的计算包括在两种工位焊接过程的路径 $wPLJ_a$ 和 $wPLG_b$, 以及机器人从起始位置到达第一个任务点和从最后一个任务点返回起始位置的路径 $iniPLJ_a$ 和 $iniPLG_b$ 。 $iniR_d$ 为机器人 R_d 的起始位置。路径的计算需要使用各工位和机器人的任务分配集合。该模型满足约束(11)~(14), 约束(11)和(12)保证每个任务都被分配到工位, 并且没有重复。约束(13)和(14)保证工位 J_2 的每个任务都被分配到机器人, 并且只分配一次。

$$\bigcup_{a=1}^3 J_aN \cup \bigcup_{b=1}^3 G_bN = SPW \cup STW \quad (11)$$

$$\sum_{a=1}^3 sJ_aN + \sum_{b=1}^3 sG_bN = |SPW| + |STW| \quad (12)$$

$$\bigcup_{d=1}^2 J_aR_dN = J_aR_dN, a = 2 \quad (13)$$

$$\sum_{d=1}^2 sJ_aR_dN = sJ_aR_dN, a = 2 \quad (14)$$

本节中的 MSMR-WTAPP 优化模型是针对图 1 所示实际生产线建立的, 模型不能直接用于优化其他的生产线。但是汽车结构件的多工位、多机器人生产线设计大都是相似的, 可以根据生产线实际组成对模型进行修改与适用。

2 改进 SPEA2 算法

2.1 编码和解码

MSMR-WTAPP 问题必须同时解决工位的焊接任务分配、机器人的焊接任务分配和机器人的焊接序列规划等子问题, 我们设计了一个包含两层染色体的编码方案来同时表示这些子问题。图 3 所示为一个包括 28 个焊点和螺柱的简化例子, 表示两层染色体的编码示意。WA 编码表示各个焊点或者螺柱的分配工位, WA 基因与分配工位的对应关系如下: 1 表示 J_1 工位, 2 表示 J_2 工位, 3 表示 J_3 工位, 4 表示 G_1 工位, 5 表示 G_2 工位, 6、7 和 8 分别表示螺柱焊枪 S_1 、 S_2 和 S_3 。图 3 所示的 WA 染色体的第 4 个位置基因为 1, 表示焊点 W_4 被分配到 J_1 工位。焊接顺序编码由 WS 染色体表示, WA 染色体的每个基因数字表示对应编号的焊点, 比如第 1 个基因 21 表示焊点 W_{21} 。WS 染色体由所有焊点编号和 -1 组成, 其中 -1 不表

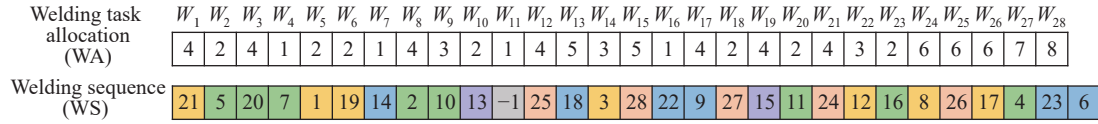


图3 编码示意图

Fig. 3 Encoding diagram

示焊点。 J_2 工位由机器人 R_1 和 R_2 共同焊接, -1 用于将分配到 J_2 工位的焊点再次分配给两个机器人。

图4示出了图3中染色体编码示例的解码过程, 解码步骤说明如下: (1) 将两条染色体 WA 和 WS 合并, WS 设置为第 1 行, WA 中与 WS 对应的分配基因放在同一列的第 2 行。同时, 将 WS 中与基因

-1 对应的 WA 基因设置为 2。(2) 根据相同的工位分配基因将染色体分组, 不改变基因顺序。(3) 根据各组内第 1 行基因的顺序, 可以得到各工位的焊接序列。比如工位 J_1 的焊接序列为 W_7 、 W_{11} 、 W_{16} 和 W_4 。因为 J_2 工位由机器人 R_1 和 R_2 共同焊接, 将 -1 左侧的序列分配给 R_1 , 右侧的序列分配给 R_2 。

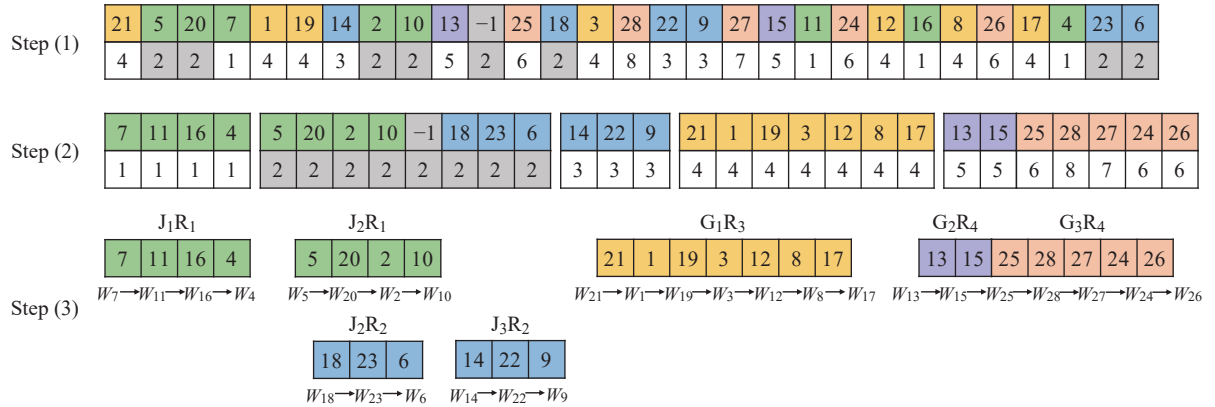


图4 WA 和 WS 解码示意图

Fig. 4 Decoding diagram of WA and WS

图2所示的实际工件包括 89 个焊点和 9 个螺柱, WA 的实际编码长度为 98, 每个位置表示序号对应编号焊点或螺柱的工位分配。WS 的实际编码长度为 99, 除所有任务的序号外还包括 -1 基因, 用来将 J_2 工位的任务分配给机器人。对于类似生产线的优化问题, 根据实际工件的焊接任务数确定 WA 的长度, 设计基因与工位的对应关系为 WA 赋予基因值。WS 编码为实际任务序号的随机顺序, 若存在双机器人焊接工位, 则再为 WS 编码增加相应基因, 用于将工位的任务再分配到机器人。设计编码后根据图4所示原理进行解码。

2.2 初始化种群

在焊接任务分配中有许多约束条件, 为了保证在种群进化过程中初始化的解和搜索到的解都是可行的, 提出了一种初始化策略。对 WA 染色体进行初始化, 根据表3和表4中的约束条件, 计算每个焊接任务可分配的工位集合 $ageneW_e$, 然后从每个任务的可分配工位基因集合中随机选择基因分配到 WA 的相应位置。WS 染色体的初始化通过随机重新排列焊点和螺柱的编号, 形成一个包含 89 个焊点

编号、9 个螺柱编号和 -1 的随机排列向量, 作为 WS 染色体。

2.3 交叉和变异算子

MSMR-WTAPP 问题的决策空间由分配编码 WA 和序列编码 WS 共同表示。对于离散决策变量, 需要根据编码类型使用不同类型的交叉算子和变异算子。在分配编码中常用的交叉算子有 4 种: 单点交叉(OPX)、两点交叉(TPX)、多点交叉(MPX)和均匀交叉(UX)。对于序列编码, 常用的交叉算子有顺序交叉(OX)、基于顺序交叉(OBX)、基于位置交叉(PBX)和部分映射交叉(PMX)。与交叉操作符相似, 不同类型的染色体编码对应不同的变异算子。本文对采用分配方案编码的 WA 染色体使用位变异操作(Bit Mutation, BM), 对序列编码的 WS 染色体使用相邻交换变异(Adjacent Swap Mutation, ASM)。本文中使用的算法都采用最优的交叉算子组合。

2.4 环境选择策略

我们提出了个体种群密度(IPD)策略用于衡量个体的多样性, 并用于改进 SPEA2 算法来求解上述的 MSMR-WTAPP 问题。IPD 策略的伪代码如表5

所示,使用目标空间中每个个体与其余个体的欧氏距离(DA)来计算 IPD。首先,计算每个个体与其他所有个体归一化后最小欧式距离的集合(minDA)。然后,将集合内的最大值作为统计种群密度的范围(range)。最后,计算每个个体的 range 内存在的个体数,作为当前个体的 IPD。

表 5 算法 1
Table 5 Algorithm 1

Algorithm 1: IPD(Individual population density)

Input: Pop(Population), DA(DistanceArray)

Output: IPD

```

1  for  $i = 1$  to PopSize do
2     $DA[i][i] = \infty$ ;
3     $minDA[i] = \infty$ ;
4    for  $j = 1$  to PopSize do
5       $DA[i][j] = distance(Pop\_i, Pop\_j)$ ;
6       $DA[j][i] = distance(Pop\_i, Pop\_j)$ ;
7      if  $DA[i][j] < minDA[i]$  then
8         $minDA[i] = DA[i][j]$ ;
9      end
10   end
11 end
12  $range = \max(minDA)$ ;
13 for  $i = 1$  to PopSize do
14    $IPD[i] = 0$ ;
15   for  $j = 1$  to PopSize do
16     if  $DA[i][j] \leq range$  then
17        $IPD[i] = IPD[i] + 1$ ;
18     end
19   end
20 end
21 return IPD

```

SPEA2+IPD 算法总体流程如图 5 所示。SPEA2+IPD 对 SPEA2 个体适应度 $F(i)$ 的计算进行了改进。外部种群 $\bar{P}$ 和种群 P 中每个个体 i 都被赋予一个强度值 $S(i)$, 表示 i 支配的个体数量。每个个体 i 原始适应度 $R(i)$ 为支配 i 的个体的强度和, 计算如下:

$$R(i) = \sum_{j \in P \cup \bar{P}, j < i} S(j) \quad (15)$$

个体 i 的适应度 $F(i)$ 由 $R(i)$ 和 IPD 来确定, 计算公式如下:

$$F(i) = R(i) + \lg(1 + IPD(i)) \quad (16)$$

$F(i)$ 计算完成后, 更新外部种群 $\bar{P}$, 首先选择 $F(i)$ 小于 1 的个体放入 $\bar{P}$ 。然后对 $\bar{P}$ 进行处理。假设种群大小为 N , 如果 $|\bar{P}|$ 等于 N , 则选择完成; 如果

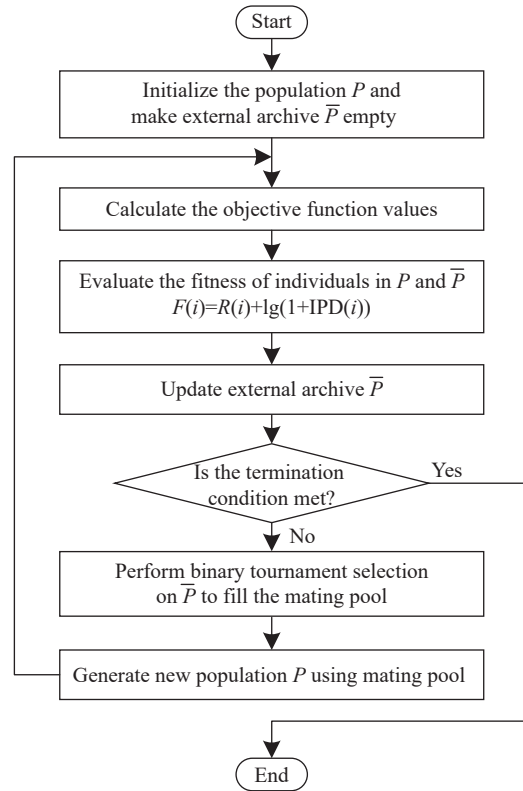


图 5 SPEA2+IPD 算法流程图

Fig. 5 Flow chart of SPEA2+IPD algorithm

$|\bar{P}|$ 小于 N , 则按照 $F(i)$ 从小到大继续选择个体放入 $\bar{P}$, 直到 $|\bar{P}|$ 等于 N ; 如果 $|\bar{P}|$ 大于 N , 则按照 $IPD(i)$ 从大到小删除 $\bar{P}$ 中的个体, 直到 $|\bar{P}|$ 等于 N 。

假设目标数为 M , 对 SPEA2+IPD 算法环境选择部分一次迭代的计算复杂度进行分析。个体种群密度的时间复杂度表示为 $O(M \cdot N^2)$, 个体适应度 $F(i)$ 的时间复杂度主要在于计算强度值, 计算 $R(i)$ 的时间复杂度也表示为 $O(M \cdot N^2)$, 所以 SPEA2+IPD 算法的时间复杂度为 $O(M \cdot N^2)$ 。我们研究的 MSMR-WTAPP 问题是将优化的结果用于长期的生产计划, 没有实时性的要求, 所以算法复杂度不会影响实际工业生产的时效。

3 实验和分析

3.1 测试实例

对于 MSMR-WTAPP 问题, 每个工位上的夹具都是专门为装配的零件特殊设计的。因此, 根据表 3 和表 4 中对焊接任务分配的约束条件, 简化一部分焊点和螺柱, 并生成 16 个实例来测试算法的性能。所有测试实例的简化都是基于生产线的组成和工件焊接的工序约束。焊点和螺柱的数量分别设置为 $N_p \in \{50, 60, 70, 80\}$ 和 $N_s \in \{0, 3, 6, 9\}$ 。每个实例都

可以被标记为“ $N_p N_s$ ”。例如,“50_9”表示该实例需要焊接 50 个焊点和 9 个螺柱。测试实例的简化方案如表 6 和表 7 所示。

表 6 测试实例焊点

Table 6 Test instances welding points

N_s	Reduced welding points set
80	$\{W_{14}, W_{16}, W_{18}, W_{20}, W_{22}, W_{24}, W_{26}, W_{30}, W_{32}\}$
70	$\{W_{34}, W_{36}, W_{38}, W_{40}, W_{44}, W_{46}, W_{48}, W_{50}, W_{52}, W_{54}\} \cup 80$
60	$\{W_{56}, W_{59}, W_{61}, W_{63}, W_{65}, W_{67}, W_{69}, W_{71}, W_{73}, W_{75}\} \cup 70$
50	$\{W_1, W_3, W_5, W_7, W_9, W_{77}, W_{79}, W_{81}, W_{83}, W_{85}, W_{89}\} \cup 60$

表 7 测试实例螺柱

Table 7 Test instances studs

N_s	Reduced studs set
0	$\{W_{90}, W_{91}, W_{92}, W_{93}, W_{94}, W_{95}, W_{96}, W_{97}, W_{98}\}$
3	$\{W_{90}, W_{92}, W_{93}, W_{94}, W_{96}, W_{97}\}$
6	$\{W_{90}, W_{93}, W_{96}\}$
9	$\emptyset$

3.2 评价指标

使用反转世代距离(Inverted Generational Distance, IGD)^[14]和超体积(Hypervolume, HV)^[15]来评价算法得到非支配解集的综合性能。IGD 值越小或 HV 值越大,说明解集的收敛性和多样性越好。

IGD 是衡量解集性能的综合度量指标,可以表示为:

$$IGD = \frac{1}{|PF^*|} \sum_{x \in PF^*} \min_{y \in PF} \text{dis}(x, y) \quad (17)$$

其中, PF^* 表示参考集, $|PF^*|$ 为 PF^* 的大小。

HV 是由解集 PF 和参考点 $r = (r_1, r_2, \dots, r_m)^T$ 组成的空间的超体积,计算方法为:

$$HV = \delta \left(\bigcup_{i=1}^{|PF|} v_i \right) \quad (18)$$

其中, m 为目标空间的维数, δ 表示用于测量体积的勒贝格测量值, v_i 表示 PF 中的第 i 个解的体积贡献值。

3.3 参数设置

参数的设置对算法的性能有很大的影响。以实际工件(89_9)为实例,研究了 SPEA2+IPD 和 3 种对比算法 SPEA2^[16]、SPEA 2+SDE^[17]和 ISDE+^[18]不同参数设置的影响。所有的算法都使用目标函数计算次数(NFEs)作为终止条件, NFEs 设置为 100 000, 每组实验都为 30 次重复实验。首先探索交叉算子组合

对算法性能的影响,种群大小设置为 200。为了区分突变算子的影响,将交叉率(P_c)设为 1, 突变率(P_m)设为 0。实验结果如图 6 所示, IGD 平均值最小的交叉算子组合表示为黑色。然后探索算法的最优种群大小(PS), 种群范围为 100~300, 间隔为 50。这一组实验采用图 6 中的最优交叉组合, 交叉率为 1, 突变率为 0。实验结果如图 7 所示, 红色圆圈表示平均 IGD 值。

最后对 P_c 和 P_m 进行研究, 实验将交叉算子和种群大小设置为图 6 和图 7 中的最优结果。实验结果如图 8 所示, 其中横轴表示交叉率, 不同颜色的列表示突变率, 纵轴表示 IGD 平均值。红色三角形表示具有最小 IGD 平均值的列。表 8 示出了后续对比实验中用于各个算法的最优参数设置。

3.4 策略有效性对比实验

为了验证 SPEA2+IPD 算法对于优化 MSMR-WTAPP 问题的有效性, 我们将其与 3 种多目标进化算法(Multi-Objective Evolutionary Algorithms, MOEAs), 即原始算法 SPEA2、SPEA2 的改进算法 SPEA2+SDE 以及 SPEA2+SDE 改进策略对应的另一种算法 ISDE+, 进行了对比。实验结果如表 9 和表 10 所示, 其中最优结果以粗体显示。

表 9 给出了 4 种算法的 IGD 平均值和 Wilcoxon 符号秩检验结果, 置信水平设置为 95%(对应于 $\alpha=0.05$)。对于 IGD 平均值, 除了“60_6”、“70_3”、“80_0”和“80_3”4 个实例, SPEA2+IPD 在其余测试实例上都最小。根据 Wilcoxon 符号秩检验的 p -value, SPEA2+IPD 在除“80_3”以外的 16 个测试实例上都显著优于原始 SPEA2 算法。与 SPEA2+SDE 相比, SPEA2+IPD 只在“60_6”1 个实例上显著落后, 在 4 个实例中与 SPEA2+SDE 没有明显差异, 在其余 12 个实例中显著优于 SPEA2+SDE。与 ISDE+相比, SPEA2+IPD 在 16 个测试实例上都显著优于 ISDE+, 只有在“80_0”实例上两种算法没有显著性差异。

表 10 描述了 HV 的统计结果。从 Wilcoxon 符号秩检验和平均 HV 值中可以看出, SPEA2+IPD 产生了 13 个最优的 IGD 值。与原始 SPEA2 算法相比, SPEA2+IPD 的 HV 平均值在所有实例上都优于 SPEA2, Wilcoxon 符号秩检验结果显示在 16 个实例上显著优于 SPEA2, 只在实例“50_3”上与 SPEA2 没有显著差异。与 SPEA2+SDE 相比, SPEA2+IPD 在“60_6”和“80_3”实例上显著落后, 在 5 个实例上两种算法没有显著差异, 在 10 个实例上 SPEA2+IPD 显著更优。SPEA2+IPD 在 13 个实例上显著优于 ISDE+, 在 2 个实例上没有表现出显著差异, 只在 2 个实例上

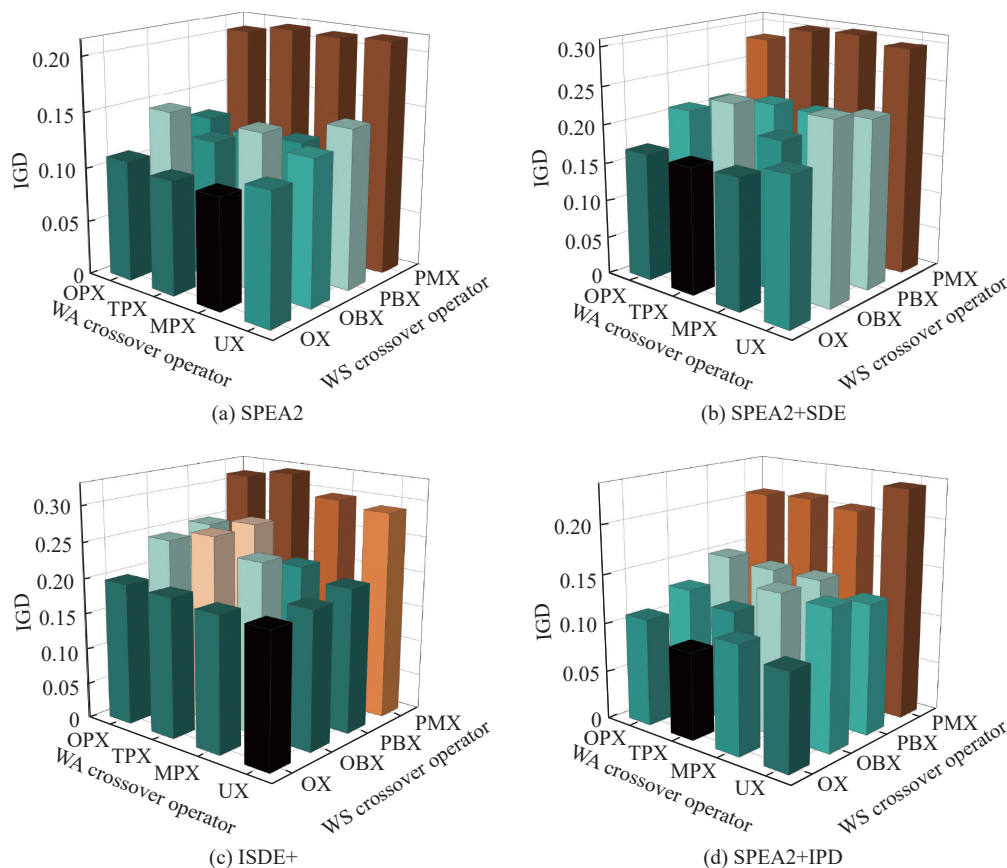


图 6 交叉算子组合设置

Fig. 6 Crossover operator combination setting

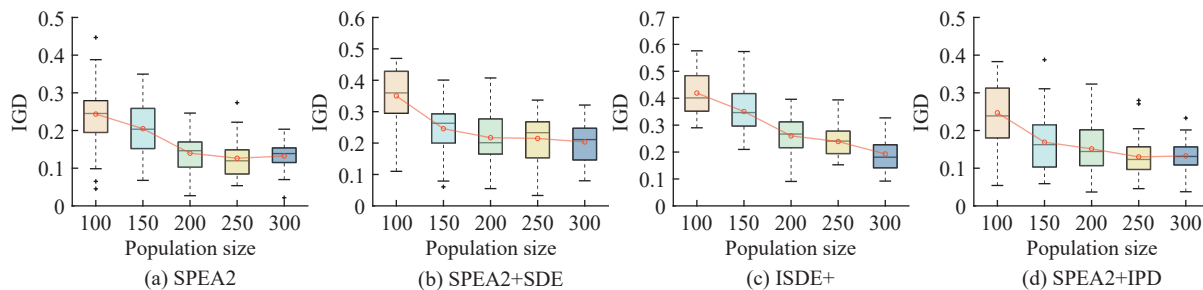


图 7 种群大小设置

Fig. 7 Population size setting

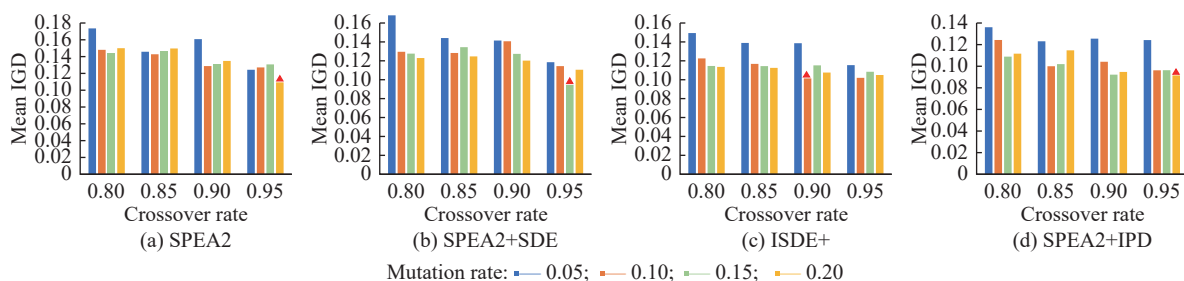


图 8 交叉、变异概率设置

Fig. 8 Crossover and mutation probability setting

落后。

从表 9 和表 10 可以明显看出, SPEA2+IPD 在两个综合指标上都明显更好, 与其他 3 种算法相比有更

大的优势。此外, 基于 Wilcoxon 符号秩检验的结果, SPEA2+IPD 在大多数情况下的效果最好。综上所述, 实验结果证实了 SPEA2+IPD 求解 MSMR-WTAPP

表 8 参数设置
Table 8 Parameter settings

Algorithm	Operator	PS	P_c	P_m
SPEA2	MPX, OX	250	0.95	0.2
SPEA2+SDE	TPX, OX	300	0.95	0.15
ISDE+	UX, OX	300	0.9	0.1
SPEA2+IPD	TPX, OX	250	0.95	0.2

表 9 IGD 平均值和 Wilcoxon 符号秩检验结果
Table 9 Mean IGD values and Wilcoxon signed rank test results

Test instance	Mean IGD				p -Value		
	SPEA2	SPEA2+SDE	ISDE+	SPEA2+IPD	SPEA2+IPD vs. SPEA2	SPEA2+IPD vs. SPEA2+SDE	SPEA2+IPD vs. ISDE+
50_0	0.0179	0.0186	0.0278	0.0155	3.03×10^{-6}	3.34×10^{-6}	1.65×10^{-6}
50_3	0.0235	0.0251	0.0348	0.0225	4.53×10^{-3}	1.56×10^{-5}	1.65×10^{-6}
50_6	0.0258	0.0252	0.0312	0.0204	1.65×10^{-6}	1.65×10^{-6}	1.65×10^{-6}
50_9	0.0394	0.0381	0.0484	0.0314	3.34×10^{-6}	1.65×10^{-6}	1.65×10^{-6}
60_0	0.0194	0.0195	0.0223	0.0153	1.65×10^{-6}	1.65×10^{-6}	1.65×10^{-6}
60_3	0.0313	0.0294	0.0403	0.0257	1.83×10^{-6}	3.34×10^{-6}	1.65×10^{-6}
60_6	0.0394	0.0315	0.0451	0.0334	3.03×10^{-6}	1.15×10^{-3}	1.65×10^{-6}
60_9	0.0390	0.0323	0.0451	0.0301	3.03×10^{-6}	1.23×10^{-1}	1.65×10^{-6}
70_0	0.0387	0.0357	0.0408	0.0335	2.24×10^{-6}	1.88×10^{-5}	1.65×10^{-6}
70_3	0.0405	0.0349	0.0419	0.0356	3.26×10^{-5}	1.81×10^{-1}	7.33×10^{-6}
70_6	0.0773	0.0555	0.0765	0.0521	1.65×10^{-6}	1.44×10^{-2}	1.65×10^{-6}
70_9	0.0466	0.0482	0.0511	0.0426	4.53×10^{-3}	1.65×10^{-6}	1.67×10^{-4}
80_0	0.0421	0.0394	0.0365	0.0370	2.97×10^{-5}	3.08×10^{-2}	6.81×10^{-1}
80_3	0.0491	0.0525	0.0608	0.0587	2.24×10^{-6}	5.08×10^{-5}	2.02×10^{-1}
80_6	0.0566	0.0550	0.0586	0.0543	2.67×10^{-3}	1.56×10^{-1}	3.27×10^{-3}
80_9	0.0678	0.0591	0.0794	0.0590	1.83×10^{-6}	8.53×10^{-1}	1.65×10^{-6}
89_9	0.1481	0.1200	0.1271	0.1071	2.02×10^{-6}	1.65×10^{-6}	4.08×10^{-6}

表 10 HV 平均值和 Wilcoxon 符号秩检验结果
Table 10 Mean HV values and Wilcoxon signed rank test results

Test instance	Mean HV				p -Value		
	SPEA2	SPEA2+SDE	ISDE+	SPEA2+IPD	SPEA2+IPD vs. SPEA2	SPEA2+IPD vs. SPEA2+SDE	SPEA2+IPD vs. ISDE+
50_0	0.2536	0.2536	0.1913	0.2817	1.65×10^{-6}	1.65×10^{-6}	1.65×10^{-4}
50_3	0.1792	0.1661	0.1000	0.1803	2.94×10^{-1}	2.95×10^{-4}	2.02×10^{-4}
50_6	0.1759	0.1835	0.1410	0.2297	1.83×10^{-6}	6.04×10^{-6}	1.65×10^{-4}
50_9	0.0803	0.0856	0.0542	0.1094	1.65×10^{-6}	1.83×10^{-6}	1.65×10^{-4}
60_0	0.2795	0.2919	0.2756	0.3305	1.65×10^{-6}	1.65×10^{-6}	1.65×10^{-4}
60_3	0.1119	0.1530	0.0813	0.1553	1.65×10^{-6}	7.58×10^{-6}	1.65×10^{-4}
60_6	0.0704	0.0897	0.0482	0.0817	2.51×10^{-4}	3.19×10^{-4}	1.65×10^{-4}
60_9	0.0719	0.0866	0.0446	0.0870	2.32×10^{-4}	9.02×10^{-1}	1.83×10^{-4}
70_0	0.0796	0.0840	0.1022	0.0989	2.72×10^{-4}	4.04×10^{-4}	3.24×10^{-1}
70_3	0.0521	0.0768	0.0768	0.0792	1.83×10^{-6}	4.34×10^{-1}	3.65×10^{-1}
70_6	0.0314	0.0877	0.0738	0.0919	1.65×10^{-6}	9.51×10^{-1}	1.65×10^{-3}
70_9	0.0835	0.0732	0.0808	0.0975	2.92×10^{-2}	1.65×10^{-6}	3.27×10^{-3}
80_0	0.1140	0.1497	0.1733	0.1494	2.72×10^{-5}	7.89×10^{-1}	6.39×10^{-4}
80_3	0.1234	0.1305	0.1451	0.0970	2.24×10^{-6}	1.83×10^{-6}	1.65×10^{-6}
80_6	0.0968	0.0958	0.1007	0.1183	1.72×10^{-5}	7.43×10^{-4}	3.97×10^{-2}
80_9	0.0593	0.0744	0.0675	0.0902	1.65×10^{-6}	1.65×10^{-3}	1.18×10^{-5}
89_9	0.0117	0.0051	0.0085	0.0133	2.72×10^{-5}	6.89×10^{-4}	2.26×10^{-5}

问题的有效性。

3.5 实际案例研究

为了可视化 4 种 MOEAs 的性能, 这些 MOEAs 得到的非支配解集如图 9 所示。可以看出, SPEA2+IPD 算法得到的解集在保证了解集收敛性的同时, 多样性显著优于其他对比算法, 在目标空间搜索到了更多并且分布更均匀的解。

图 9(a) 示出了具有 3 个目标的三维视图。从

图 9(b)、9(c) 和 9(d) 可以看出, 3 个目标互相都存在冲突关系。在图 9 中, 点 A、B 和 C 分别表示由 SPEA2+IPD 得到的非支配解集内 T_{first} 、 $T_{\text{subsequence}}$ 和 PL 这 3 个目标的最小值。表 11 记录了 A、B、C 点的 3 个目标函数值及工厂人工优化的结果。与人工规划结果相比, A 点的第 1 个工件加工时间 (T_{first}) 下降了 22.8%, B 点的 $T_{\text{subsequence}}$ 目标减少了 15.3%, C 点机器人路径长度降低了 13.6%。

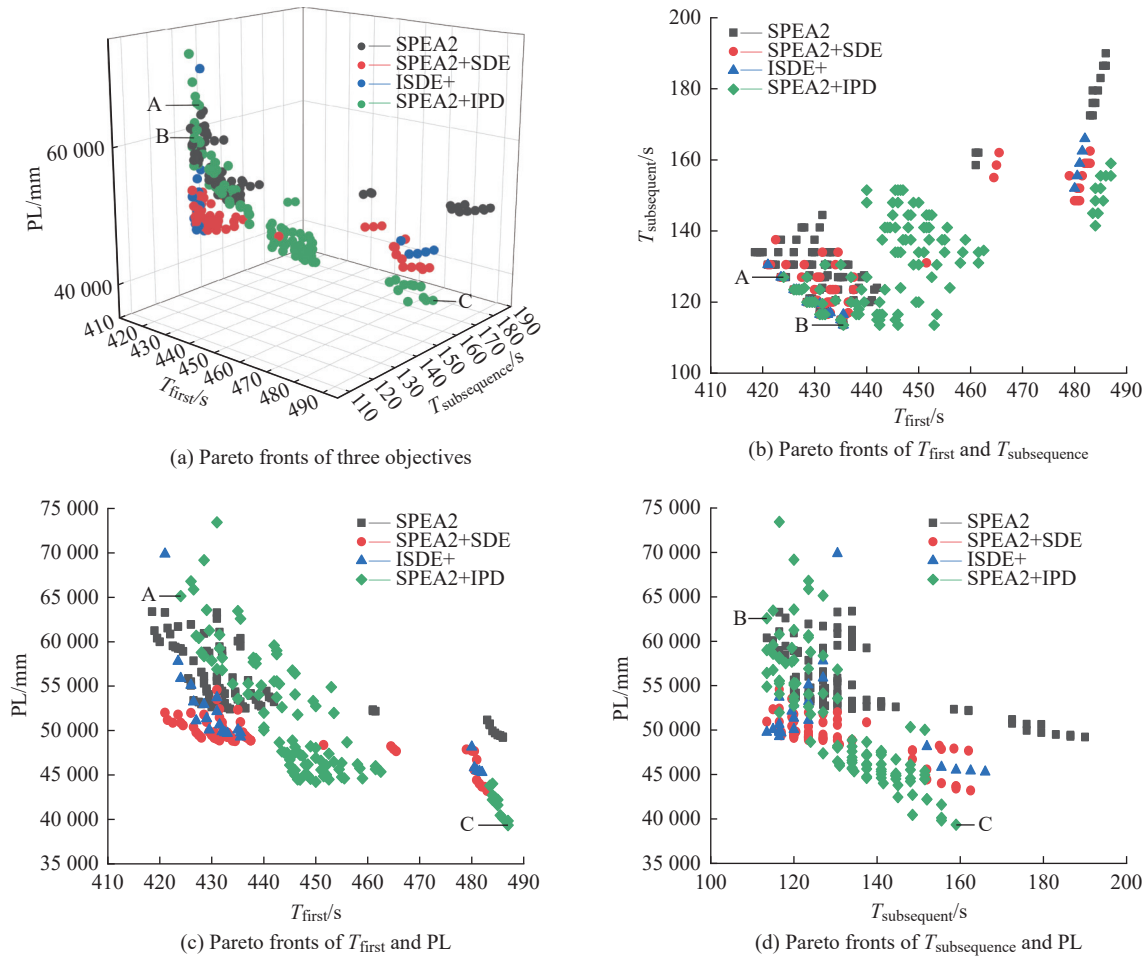


图 9 4 种算法得到的 Pareto 前沿

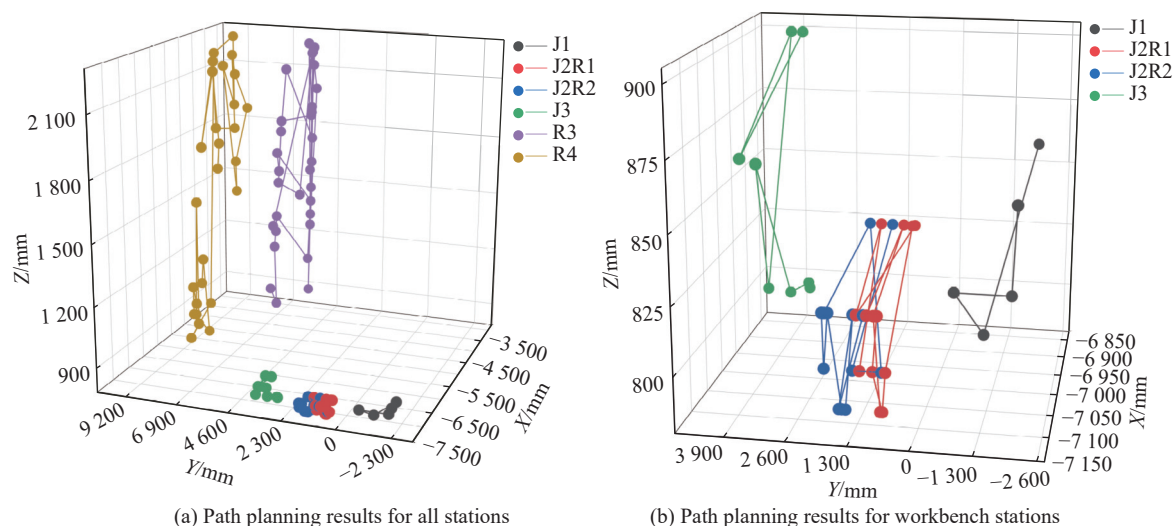
Fig. 9 Pareto fronts obtained by four algorithms

表 11 SPEA2+IPD 得到最优解和工厂优化结果

Table 11 SPEA2+IPD obtained optimal solutions and factory result

Solution	$T_{\text{first}}/\text{s}$	$T_{\text{subsequence}}/\text{s}$	PL/mm
Point A	424	127	65 131.48
Point B	435.5	113.5	62 576.83
Point C	487	159	39 347.51
Factory result	549.5	134	45 515.12

MSMR-WTAPP 问题的特点是相同工件连续生产, 减小 $T_{\text{subsequence}}$ 是提高生产效率的关键因素。在大量工件不间断生产的情况下, 降低 $T_{\text{subsequence}}$ 可以更加有效地提高生产效率。图 9 所示 B 点为非支配解集 $T_{\text{subsequence}}$ 目标的最优点, B 点的目标值表明, 与工厂人工优化后实际生产节拍相比, 第 1 个工件生产时间缩短 20.7%, 后续每个工件的时间都缩短了 15.2%。B 点的路径规划结果如图 10 所示, 按照 B 点优化的方案进行焊接, 可以很大程度提高工厂的生产效率。

图 10 目标 $T_{\text{subsequence}}$ 最优结果的焊接路径Fig. 10 Welding path for the result with the best $T_{\text{subsequence}}$

4 结 论

针对多目标 MSMR-WTAPP 问题, 提出了一种改进 SPEA2 算法。首先, 建立了考虑生产线和工件等多种约束的多目标优化数学模型, 给出了约束条件和优化目标的计算方法。优化目标分别是生产第一个工件加工时间、后续每个工件加工需要的时间和不考虑碰撞的机器人路径长度。然后, 设计了分配和序列的两层编码方式和解码过程, 研究了机器人工序无冲突的解码方案, 并提出了 SPEA2+IPD 算法用于求解该问题。最后, 把实际工件以及简化焊点与螺柱产生的 16 个实例作为测试对象, 将 SPEA2+IPD 算法与其他经典算法进行比较, 实验结果表明 SPEA2+IPD 在优化 MSMR-WTAPP 问题上优于其他算法, 与工厂人工优化的结果相比, SPEA2+IPD 可以大幅提高生产效率, 对工厂的规划与调度具有指导价值。

参考文献:

- [1] WANG B, HU S, SUN L, *et al.* Intelligent welding system technologies: State-of-the-art review and perspectives[J]. *Journal of Manufacturing Systems*, 2020, 56: 373-391.
- [2] LIANG S, RAJORA M, LIU X, *et al.* Intelligent manufacturing systems: A review[J]. *International Journal of Mechanical Engineering and Robotics Research*, 2018, 7(3): 324-330.
- [3] 汤彬, 王学武, 薛立卡, 等. 双焊接机器人避障路径规划[J]. *华东理工大学学报 (自然科学版)*, 2017, 43(3): 417-424.
- [4] 王学武, 时应盼, 顾幸生, 等. 基于分区粒子群算法的焊接机器人路径规划[J]. *华东理工大学学报 (自然科学版)*, 2014, 40(5): 95-101.
- [5] JIANG L, LIU S, CUI Y, *et al.* Path planning for robotic manipulator in complex multi-obstacle environment based on Improved_RRT[J]. *IEEE/ASME Transactions on Mechatronics*, 2022, 27(6): 4774-4785.
- [6] SPENSIERI D, CARLSON J, EKSTEDT F, *et al.* An iterative approach for collision free routing and scheduling in multirobot stations[J]. *IEEE Transactions on Automation Science and Engineering*, 2015, 13(2): 950-962.
- [7] SUÁREZ-RUIZ F, LEMBONO T, PHAM Q. RoboTSP-A fast solution to the robotic task sequencing problem[C] // 2018 IEEE International Conference on Robotics and Automation (ICRA). Brisbane: IEEE, 2018: 1611-1616.
- [8] FENG Z, HU G, SUN Y, *et al.* An overview of collaborative robotic manipulation in multi-robot systems[J]. *Annual Reviews in Control*, 2020, 49: 113-127.
- [9] CHAKRAA H, GUÉRIN F, LECLERCQ E, *et al.* Optimization techniques for multi-robot task allocation problems: Review on the state-of-the-art[J]. *Robotics and Autonomous Systems*, 2023, 168: 104492.
- [10] WANG X, YAN Y, GU X. Spot welding robot path planning using intelligent algorithm[J]. *Journal of Manufacturing Processes*, 2019, 42: 1-10.
- [11] JIN L, LI S. Distributed task allocation of multiple robots: A control perspective[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018, 48(5): 693-701.
- [12] TOUZANI H, HADJ-ABDELKADER H, SÉGUY N, *et al.* Multi-robot task sequencing & automatic path planning for cycle time optimization: Application for car production line[J]. *IEEE Robotics and Automation Letters*, 2021, 6(2): 1335-1342.

- [13] ZHAO W, LIU Y, WANG Y, *et al.* Distributed spot welding task allocation and sequential planning for multi-station multi-robot coordinate assembly processes[J]. The International Journal of Advanced Manufacturing Technology, 2023, 127(11/12): 5233-5251.
- [14] ZITZLER E, THIELE L, LAUMANN M, *et al.* Performance assessment of multi-objective optimizers: An analysis and review[J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 117-132.
- [15] ZITZLER E, THIELE L. Multi-objective evolutionary algorithms: A comparative case study and the strength pareto approach[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257-271.
- [16] ZITZLER E, LAUMANN M, THIELE L. SPEA2: Improving the strength pareto evolutionary algorithm[C] // Evolutionary Methods for Design, Optimization and Control with Application to Industrial Problems. Athens, Greece: International Center for Numerical Methods in Engineering, 2001: 1-21.
- [17] LI M, YANG S, LIU X. Shift-based density estimation for pareto-based algorithms in many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2013, 18(3): 348-365.
- [18] PAMULAPATI T, MALLIPEDDI R, SUGANTHAN P. ISDE+: An indicator for multi and many-objective optimization[J]. IEEE Transactions on Evolutionary Computation, 2019, 23(2): 346-352.

Multi-Station Multi-Robot Welding Task Assignment and Path Planning

WANG Ye, WANG Xuewu, GU Xingsheng

(Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China)

Abstract: In the automotive industry, optimizing the production process of manufacturing identical products is the key to improving production efficiency. Robot automated production lines are an ideal choice for modern manufacturing. The existing research mainly focuses on the finite subproblems of production lines. This paper considers the welding task assignment and welding sequence planning of multi-station and multi-robot production line in the actual automobile manufacturing industry, in which many constraints of workpiece and production line composition are involved. A mathematical model of Multi-Station Multi-Robot Welding Task Assignment and Path Planning (MSMR-WTAPP) is established, whose optimization objective is to minimize the first workpiece processing time, the subsequent workpiece processing time, and the robot motion path length synchronously. An improved SPEA2 algorithm based on individual population density (SPEA2+IPD) is proposed and a dual layer coding scheme is designed for the coupling problem of welding task assignment and welding sequence planning, and the decoding process of robot operation is addressed. Finally, the effectiveness and superiority of SPEA2+IPD algorithm in optimizing the cycle time, efficiency and welding path of multi station and multi robot production lines are verified via simulation experiments. Compared to the actual production cycle of the factory, the production cycle optimized by SPEA2+IPD algorithm is 20.7% shorter for the first workpiece, and the production time of each subsequent workpiece is reduced by 15.2%. This indicates that the proposed model and algorithm have practical significance for optimizing the factory production.

Key words: automobile manufacturing; industrial robot; welding; task assignment; path planning

(责任编辑: 李娟)