

DOI:10.13196/j.cims.2017.11.012

多重资源约束下的多目标 U 型装配线平衡

张子凯, 唐秋华⁺, 张利平

(武汉科技大学 机械自动化学院, 湖北 武汉 430081)

摘 要:为处理因设备故障、订单变化等引起的任务量波动或生产中断问题,在关键工序设置多个并行可选设备、在生产子线设置助理,以保证装配线的生产率。针对该类问题,构建随机工时下基于资源分配的成本、效率双目标 U 型装配线平衡模型,并采用 Benders 分解法,将问题分解为设备和助理分配主问题、工序分配子问题,以降低模型求解的复杂度。提出基于 Benders 分解的快速非支配遗传算法,通过三层编码及解码来适应多决策变量;采用非回溯的 Pareto 层级构造和拥挤距离,实现种群评价与选择;提出基于概率的层次化遗传操作,以扩充邻域结构、增强寻优能力、避免局部优化。通过非支配解比率、Pareto 前沿收敛性和个体间距离度量指标分析所提算法、多目标遗传算法和非支配排序遗传算法,证明算法获得了逼近 Pareto 最优前沿的非支配解集,且具有良好的收敛性和分布性。

关键词:U 型装配线;资源分配;多目标优化;快速非支配遗传算法

中图分类号:TH166;TP18

文献标识码:A

Multi-objective U-shaped assembly line balancing under multi-resource restriction

ZHANG Zikai, TANG Qiu-hua⁺, ZHANG Liping

(School of Machinery Automation, Wuhan University of Science and Technology, Wuhan 430081, China)

Abstract:To handle the production disruptions and workload fluctuations caused by equipment failure and order's change, the multiple parallel optional equipment in key process and assistant in production line were set up. For decision variables, the benders decomposition method was adopted to divide the origin problem into the main problem on allocating equipments and assistants and the sub-problems on assign tasks, which reduced the complexity of model solution; the Benders-based rapid Non-dominated Sorting Genetic Algorithm (BNSGA-II) was put forward with three layers of coding and decoding to adapt to decision variables; the hierarchy structure and crowding distance of non-backtracing Pareto method were used to implement population evaluation and selection; the hierarchical genetic operator based on probability was proposed to widen the neighborhood structure, improve the optimization ability and avoid the local optimization. The proposed algorithm was analyzed by ratio of non-dominated solution, convergence of Pareto frontier solution and spread metric, and the result showed that the algorithm could obtain the approximate non-dominated solutions of Pareto optimal frontier and had good convergence and diversity by a comparison with MOGA and NSGA.

Keywords: U-shaped assembly line; resources assignment; multi-objective optimization; fast elitist non-dominated sorting genetic algorithm

收稿日期:2016-07-06;修订日期:2016-09-16。Received 06 July 2016; accepted 16 Sep. 2016.

基金项目:国家自然科学基金资助项目(51275366,51305311);中国博士后科学基金资助项目(20134219110002,2013M542073)。Foundation items:Project supported by the National Natural Science Foundation, China(No. 51275366,51305311), and the China Postdoctoral Science Foundation, China(No. 20134219110002,2013M542073).

0 引言

装配线平衡问题指在满足工序间优先关系约束的前提下,将装配线上的工序分配到工位中,并且工位中工序的加工时间之和不超过节拍。由于存在优先关系,随着工序规模的增加,装配线平衡问题的困难系数呈指数增长,该问题也是 NP-hard 问题。如今,随着我国全球贸易合作的增加,订单的需求变化从按库存生产(Make to Stock, MTS)模式逐渐向按订单制造(Make to Order, MTO)模式转化,需要高柔性化的装配线。同时,随着对资源分配的重视,装配线的研究目标已不仅局限于效率和工作负载,最小化成本成为当前主要的目标。汽车、电子等行业也越来越注重这方面的研究。

U 型装配线于 1994 年首次被 Miltenburg^[1]提出,其比传统的直线型装配线柔性更高、效率也更高。Miltenburg 介绍了 U 型装配线的特点,阐述了 U 型装配线的优点,并构建了 U 型装配线简单的数学模型。随后,众多学者开始采用整数规划方法来求解 U 型装配线。Urban 等^[2]采用整数规划方法实现了 U 型装配线的求解;Scholl 等^[3]采用分支定界法求解 U 型装配线平衡问题。也有学者采用启发式和元启发式算法来求解该问题,例如:Erel 等^[4]提出模拟退火算法,Gokcen 等^[5]采用最短路径法,Hwang 等^[6]提出遗传算法,Sabuncuoglu 等^[7]采用蚁群算法,张子凯等^[8]设计了改进型遗传算法。

目前,为适应工业的需要,资源分配问题逐渐应用到 U 型装配线中。Manavizadeh 等^[9]在 U 型装配线问题中同时考虑了学习效应和人力分配,首次提出永久工人和暂时工人两种操作工人;Kara 等^[10]和 Jayaswal 等^[11]采用设备与助理人员帮助操作工人工作,从而降低操作时间。以上文献在资源问题中仅考虑单一目标,无法全面体现资源问题的特殊性。

随着准时生产(Just in Time, JIT)模式在制造业的流行,企业在决定装配线布局时,不是依靠单一目标做出决策,而是进行多目标综合判断。装配线领域中的目标为最小化工位、最小化节拍、最大化线效率、最小化工位负载、最小化成本等。Kara 等^[12]在 U 型装配线的平衡排序问题中采用工位负载、最小单元使用率和安装成本作为目标函数,并采用加权的方式对多目标进行归一化处理;Rabbanian 等^[13]针对混合 U 型装配线第一类问题,采用工位负载和线效率评判结果

的优劣,利用加权的方式进行归一化处理;唐秋华等^[14]在考虑多目标问题时同样采用归一化处理方式。采用加权方式处理多目标,权值的设置会影响最终结果,而且不同量纲的目标采用加权处理不具有科学性,需要采用 Pareto 非支配求解方法。

基于以上研究,本文构建随机工时下、基于资源分配的成本、效率双目标 U 型装配线平衡模型,并采用基于 Benders 分解的快速非支配遗传算法(Benders fast elitist Non-dominated Sorting Genetic Algorithm, BNSGA-II)实现模型的求解。

1 问题陈述及模型建立

1.1 假设条件

基于资源分配的 U 型装配线平衡问题不仅要分配工序,还要将操作工人和设备分配到工位中,因此需满足以下假设:

- (1)该装配线上可装配多产品,满足多种需求。
- (2)每个工序操作均不可再分,且每个工序必须分配一次。
- (3)工序间的优先关系已知且确定。
- (4)每个工序都需要操作工人,部分工序需要额外的设备或助理。
- (5)工序操作时间确定,但是在设备和助理的帮助下,操作时间会减小。
- (6)设备不能从出口线移动到入口线,工人和助理可以来回移动。
- (7)操作工人的数量等于工位数量,设备和助理的数量有限。

1.2 符号说明

- I 为操作集, $I = \{1, 2, 3, \dots, I\}$, $i, r, s \in I$;
 J 为工位集, $J = \{1, 2, 3, \dots, J\}$, $j \in J$;
 E 为设备种类集合;
 W 为优先关系集,若 $(r, s) \in W$,则 r 为 s 的直接前序, s 为 r 的直接后序;
 CT 为节拍;
 NE_e 为设备 e 可以使用的数量;
 NE 为设备种类的数量;
 NA 为助理的数量;
 t_i 为操作 i 的加工时间;
 t_{iea} 为操作 i 在设备 e 上,在助理帮助下的加工时间;
 t_{ie0} 为操作 i 在设备 e 上,无助理帮助下的加工时间;

t_{i0} 为操作 i 在没有助理和设备情况下的加工时间;

CA 为使用一个助理的成本;

CE_e 为使用设备 e 的成本;

CR 为开设一个工位的成本;

X_{ij} 为 0-1 变量,若操作 i 在工位 j 的入口线则为 1,否则为 0;

Y_{ij} 为 0-1 变量,若操作 i 在工位 j 的出口线则为 1,否则为 0;

Z_j 为 0-1 变量,若工位 j 使用助理则为 1,否则为 0;

EQB_{je} 为 0-1 变量,若设备 e 分配在工位 j 的入口线则为 1,否则为 0;

EQS_{je} 为 0-1 变量,若设备 e 分配在工位 j 的出口线则为 1,否则为 0。

1.3 目标函数

目标函数为最小化成本和最大化线效率。

基于资源分配的 U 型装配线平衡问题的成本主要有工位成本、助理成本和设备成本。最小化成本的计算公式为

$$\min Cost = \min \sum_{j=1}^J (CR + Z_j \times CA) + \sum_{j=1}^J \sum_{e=1}^{NE} (EQB_{je} + EQS_{je}) \times CE_e. \quad (1)$$

装配线的线效率为各工位有效工作时间与给定时间的比值。最大化线效率的计算公式为

$$\max Efficiency = \max \frac{1}{J \times CT} \times \sum_{i=1}^I t_i \times 100\%. \quad (2)$$

1.4 U 型装配常规约束

为提高空间利用率,装配线布局有直线型、S 型、L 型、U 型等。将工序分配到工位时只考虑直接前序,依次进行分配的装配线为直线型装配线。不同于直线型装配线,U 型装配线在分配工序时,既考虑直接前序,又考虑直接后续,由入口线和出口线两部分构成。为实现某个零件的装配,需按照图 1a 所示的优先关系完成 9 个装配任务,图 1b 所示为采用具有 U 型布局的直线型装配方案,图 1c 所示为 U 型装配线。因此 U 型装配线需考虑工序分配约束、工位分配约束、优先关系约束和节拍约束。

(1) 工序分配约束 每一个工序都必须分配,可分配到入口线或者出口线,但只能分配一次。

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

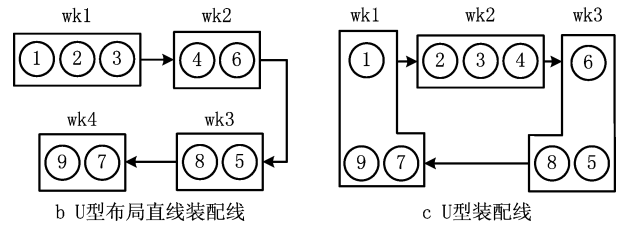


图1 U型装配线与U型布局直线型装配线

(2) 工位分配约束 每一个开设的工位都必须存在工序,且至少存在一个工序。

$$\sum_{i=1}^I X_{ij} + Y_{ij} \geq 1, \forall j \in J. \quad (4)$$

(3) 优先关系约束 当操作的所有直接前序或所有直接后序都分配完毕,该工序才能分配。

$$\sum_{j=1}^J [j \times (X_{rj} - X_{sj}) + (2 \times J - j) \times (Y_{rj} - Y_{sj})] \leq 0, \forall (r, s) \in W. \quad (5)$$

(4) 节拍约束 每一个工位的总操作时间不能超过节拍。

$$\sum_{i=1}^I [t_i \times (X_{ij} + Y_{ij})] \leq CT, \forall j \in J. \quad (6)$$

1.5 设备及助理资源约束

在汽车、电子等企业的 U 型装配生产线上,同一个工序可以由不同的设备进行操作,同一个设备也可以加工多个工序。在工位中配置不同的设备,会导致工序的加工时间不同,因此合理地分配设备能够有效减少工位数量。U 型装配线上的助理类似于生产组长或线长,具有多项技能,一般一条子线有有限个助理,可辅助所在生产线上任意工位的常规工人,分担常规工人的任务、减少装配工位数、加快各工位的操作速度,从而提高生产效率。因此考虑资源分配的 U 型装配线平衡问题也需要考虑设备分配约束、助理分配约束、设备和助理数量约束。

(1) 设备分配约束 每一个工位入口线和出口线上的操作都需要对应设备进行装配,但是一台设备不能同时进行同一工位入口线和出口线的操作。

$$\sum_{e=1}^{E_i} EQB_{je} \geq X_{ij}, \forall i \in I, \forall j \in J; \quad (7)$$

$$\sum_{e=1}^{E_i} EQS_{je} \geq Y_{ij}, \forall i \in I, \forall j \in J. \quad (8)$$

(2) 助理分配约束 将助理分配到工位中。

$$Z_j = \begin{cases} 1, & \text{工位 } j \text{ 分配到一个助理;} \\ 0, & \text{工位 } j \text{ 没有分配到助理。} \end{cases} \quad \forall j \in J. \quad (9)$$

(3)设备数量约束 设备的数量不能超过已有的设备总数。

$$\sum_{j=1}^J (EQB_{je} + EQS_{je}) \leq NE_e, \forall e \in E. \quad (10)$$

(4)助理数量约束 使用的助理数量不能超过助理总数量。

$$\sum_{j=1}^J Z_j \leq NA. \quad (11)$$

(5)操作加工时间 同一操作在不同设备上的加工时间不同,同一操作在同一设备上,有助理比没有助理的加工时间短,因此如果工位上存在多个设备可以加工同一操作,则取最短的加工时间作为操作的加工时间。

$$t_i = \min \{ t_{ica} \times (EQB_{je} \times X_{ij} \times Z_j + EQS_{je} \times Y_{ij} \times Z_j), t_{ie0} \times [EQB_{je} \times X_{ij} + EQS_{je} \times Y_{ij}] \}, \forall j \in J, \forall e \in E. \quad (12)$$

2 基于 Benders 分解的快速非支配解遗传算法

多重资源约束的装配线平衡问题涉及多个决策变量,若采用常规算法,则会在可行解的构造上增加难度。为降低编码解码的复杂度,在解码阶段采用 Benders 分解法思想,先分配设备和助理再分配工序,确定操作时间,将多个决策变量分割开来。改进的非支配排序遗传算法 (Non-dominated Sorting Genetic Algorithm II, NSGA-II) 的种群由 Pareto 前沿解构成,其种群多样性差,容易陷入局部优化。为此,采用非回溯^[15]的 Pareto 层级构造法快速实现解的分层。种群构造中,不仅涵盖前沿解,也包含支配解,以保证种群的多样性。同时,提出新的邻域结构——助理变异、设备变异、助理交叉和设备交叉,来增强算法的寻优能力。所提算法的伪代码如图 2 所示。

2.1 工序、设备和助理三层编码

考虑资源分配问题,提出一种新的编码方式——三层编码。其中第 1 层表示工序分配顺序,称为工序编码;第 2 层表示设备分配到工位问题,称为设备编码;第 3 层表示助理分配到工位问题,称为助理编码。工序编码采用 Rabbani^[13]编码方式,基于启发式规则产生序列,利用启发式规则进行工序编码可以提高算法效率。设备编码长度为 $M \times NE$, M 表示工位, NE 表示设备种类;设备编码的每一个位置填写 0-3 变量,3 表示该设备同时存在于出口线和入口线,2 表示该设备在出口线,1 表示该

Algorithm: 基于 Benders 分解的改进型非支配解遗传算法 复杂度
Input: U 型装配线数据、设备和助理资源数据
Output: Pareto 前沿解
Begin
编码阶段 $O(m \times n)$
工序、设备和助理编码;
解码阶段 (基于 Benders 分解法的解码) $O(M \times m) + O(m \times n) + O(2M) = O(m \times n)$
分配设备和助理到工位; $O(M \times m)$
确定操作时间, 分配工序; $O(m \times n)$
求解目标函数; $O(2M)$
解码结束
选择阶段 $O(m \times (m-1)/2) + O(m) = O(m \times m)$
非回溯的 Pareto 层级构造方法对初始种群进行层级构造 $O(m \times (m-1)/2)$
计算拥挤度; $O(m)$
更新初始种群;
选择结束
For $i=1$ to CC (迭代次数) $O(m \times m \times n)$
基于概率的层次化交叉/变异; $O(m)$
解码、选择; $O(m \times n)$
end For
 $T(BNSGA-II) = O(m \times n) + O(m \times n) + O(m \times m) + O(m \times m \times n) = O(m \times m \times n)$
(m : 种群规模; n : 工序数; M : 工位数)

图2 算法运算流程及复杂度

设备在入口线,0 表示不采用。助理编码长度为 M , 同样采用 0-1 变量,0 表示不采用助理,1 表示采用助理。

考虑到本文求解的是第一类装配线平衡问题,工位的数量未知,故

$$M = \lceil (2 \times \sum_i t_{i0}) / CT \rceil, \quad (13)$$

从而将工位数控制在工位下限值的 2 倍以内,排除了恶劣解。图 3 所示为图 1 中装配线的一个三层编码图。

工序编码									设备编码			助理编码		
1	2	3	4	9	8	7	6	5	$j=1$	$j=2$	$j=3$	$j=1$	$j=2$	$j=3$
1	2	3	4	9	8	7	6	5	1	2	0	1	3	0
									$e=1$	$e=2$	$e=1$	$e=2$	$e=1$	$e=2$

图3 三层编码图

假设图 1 所示的装配线计算的 $M=2$ (图 3 中 j 表示工位号), 设备种类为两种 (即 $e=1, 2$), 则图 3 中的编码表示: 在工序分配中, 第 1 个工序按照启发式规则 1 进行分配, 第 2 个工序按照启发式规则 2 进行分配, 第 3 个工序按照启发式规则 3 进行分配, 以此类推。

在设备分配中, 工位 1 的入口线分配设备 1, 出口线分配设备 2; 工位 2 的入口线分配设备 2, 出口线不分配; 工位 3 的入口线分配设备 1, 出口线分配设备 1。在助理分配中, 工位 1, 3 分配助理, 工位 2 不分配助理。

2.2 解码及修正

解码阶段不仅要 will 将工序分配到工位中, 确定工

位数,还要得到设备和助理的分配状况。该模型涉及 $X_{ij}, Y_{ij}, Z_j, EQB_{je}, EQS_{je}$ 5 个决策变量,属于大规模混合非线性优化问题,采用普通解码方式的计算复杂度高、效率低,难以求得满意解。而采用 Benders 分解法,将原问题转化成主问题和子问题,可以降低决策变量之间相互关系所带来的复杂性,提高求解速率。采用 Benders 分解法时,主问题的决策变量为 Z_j, EQB_{je}, EQS_{je} ,子问题的决策变量为 X_{ij}, Y_{ij} 。并且主问题的目标函数为式(1),约束条件为式(7)~式(11);子问题的目标函数为公式(2),约束条件为式(3)~(6)、式(12)。

具体编码步骤如下:

步骤 1 根据优先关系构建可选操作集(分为入口线操作集和出口线操作集),令 $i=1$ 。

步骤 2 根据工序编码中第 i 个位置的规则,在可选操作集中选择工序,放入工序解码染色体的第 i 个位置。

步骤 3 在可选操作集中删除所选的工序,添加可分配工序。

步骤 4 $i=i+1$,若 $i \leq I$,则转步骤 2。

步骤 5 工位 $k=1, i=1$,设备编码记为 $Ecode_{ke}$ (e 表示第 e 种设备),助理编码记为 $Acode_k$ 。

步骤 6 计算设备 e 已使用的数量,若超过规定数量,则令设备编码中 $Ecode_{ke}=0$ 。

步骤 7 计算助理已使用数量,若超过规定数量,则助理编码中 $Acode_k=0$ 。

步骤 8 计算解码染色体上 i 位置的操作时间 t_i :解码染色体上 i 位置的工序为 i' ,若工序 i' 分配在入口线,则 $t_i = \min\{Acode \times t_{jea}, t_{je0} | Ecode_{ke}=1, 3\}$;若工序 i' 分配在出口线,则 $t_i = \min\{Acode \times t_{jea}, t_{je0} | Ecode_{ke}=2, 3\}$ 。

步骤 9 若满足节拍约束,则将工序 j 分配到工位 k 中,否则转步骤 10。

步骤 10 $k=k+1, i=i-1$ 。

步骤 11 $i=i+1$,若 $i \leq I$,则转步骤 6,否则结束。

步骤 1~步骤 4 得到工序的分配顺序,步骤 5~步骤 7 将不可行的设备编码和助理编码转化成可行,步骤 8~步骤 11 得到工位及工序分配到工位的状况。

2.3 基于 Pareto 前沿的选择

2.3.1 Pareto 层级构造

Pareto 解又称非支配解。当一个解在某个目

标或某几个目标上优于另一个解,而在其他目标上较差时,这两个解都称为 Pareto 解。在构造 Pareto 层级时,算法采用非回溯的 Pareto 层级构造方法^[15]。基于双目标问题,该方法的基本步骤如下:

步骤 1 建立构造集,将初始种群放入构造集中,记当前层级为 $c=1$ 。

步骤 2 构造集中的个体数量为 I_g ,当前比较个体为构造集中的第一个个体 $i=1$ 。

步骤 3 用比较个体与构造集中的其他个体进行比较,若比较个体能够支配构造集中的其他个体,则将其他个体从构造集中剔除,否则将比较个体从构造集中剔除;若比较个体不能被任何其他个体支配,则将比较个体纳入当前层级的非支配解集中,并从构造集中剔除。

步骤 4 $i=i+1$,若构造集不为空,则转步骤 3。

步骤 5 将层级 c 中的非支配解集按照其中某一目标大小顺序进行排序。

步骤 6 将层级 c 中的非支配解集从初始种群中剔除,并将初始种群放入构造集中。

步骤 7 $c=c+1$,若构造集不为 0,则转步骤 2,否则结束。

2.3.2 基于计算距离的拥挤度评价

同一层级中,个体的拥挤度为与其相邻两个个体的距离之和:

$$E_i = |Efficiency_i - Efficiency_{i+1}| + |Efficiency_i - Efficiency_{i-1}| / \max Efficiency - \min Efficiency; \quad (14)$$

$$C_i = \frac{|Cost_i - Cost_{i+1}| + |Cost_i - Cost_{i-1}|}{\max Cost - \min Cost}; \quad (15)$$

$$Crowd_i = E_i + C_i. \quad (16)$$

式中: $Crowd_i$ 为个体的 i 拥挤度; E_i 为个体 i 的线效率的相对距离之和; C_i 为个体 i 的线效率的相对距离之和; $\max Efficiency, \min Efficiency$ 为个体种群中线效率的最大值和最小值; $\max Cost, \min Cost$ 为个体种群中成本的最大值和最小值。若个体为层级的第一个或最后一个,则 $Crowd_i=4$ 。

2.3.3 选择

采用 Pareto 层级构造法对初始种群进行层级划分,选择 M_q 个前沿解放进初始种群,以保证种群基因的优越性;在支配解中选取 M_s 个放入初始种群,以保证种群基因的多样性,避免陷入局部优化。若前沿解的个数少于 M_q ,则将前沿解全部放入初始

种群,否则删除前沿解中拥挤度最小的若干解,直到前沿解的个数等于 M_q ,以保证种群的分布性。

2.4 基于概率的层次化交叉与变异

由于研究问题的特殊性,提出新的邻域结构——设备(助理)交叉和设备(助理)变异,通过采用基于概率的层次化方法结合多种交叉(变异)方法来增强算法的寻优能力,避免算法陷入局部优化。

以基于概率的层次化交叉为例,设置工序交叉概率 LC 、设备交叉概率 EC 、助理交叉概率 AC 、第 i 个解的工序交叉概率 LCp_i 、第 i 个解的设备交叉概率 ECp_i 和第 i 个解的助理交叉概率 ACp_i 。每一代种群中的个体交叉步骤如图 4 所示。

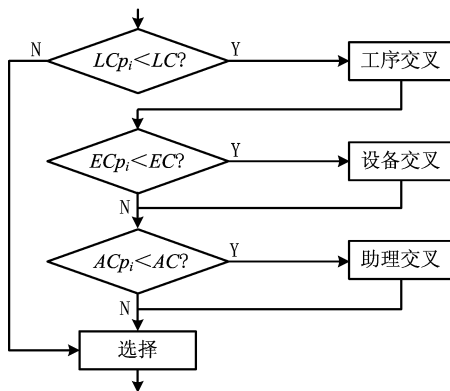


图4 种群中的个体交叉步骤

工序交叉是在两条母体染色体工序编码上选择两个交叉点,将两个交叉点之间的编码规则交换后得到两条新的子代编码染色体;工序变异在母体染色体工序编码中选择两点,重新生成两点间的工序编码。

设备(助理)交叉在母体染色体的设备(助理)编码中选择两个工位点,交换两工位之间所有工位中设备(助理)的分配编码,得到两条新的染色体;设备(助理)变异在设备(助理)编码中选择两个工位点,重新生成两工位之间所有工位中设备(助理)的编码,得到新的染色体。

3 实例分析及讨论

3.1 实例说明

实验在个人电脑(处理器为 Intel(R) Core(TM) i5 CPU 760 @ 2.80 GHz 2.79 GHz,系统为 windows 7 旗舰版)上运行,实验数据采用 U 型装配线标杆案例(数据来源为装配线平衡专业网站 <http://alb.mansci.de/>),设备资源数据采用 Sachin^[11]提出的方法生成:

$$NE_e \sim U(2,4), \forall e \in \{E:e \neq 0\}; \quad (17)$$

$$t_{ie0} = (100 - d)\% \times t_{i00}, \forall e \in \{E:e \neq 0\},$$

$$d \sim U(20,50); \quad (18)$$

$$t_{ieu} = (100 - d)\% \times t_{ie0}, \forall e \in \{E:e \neq 0\},$$

$$d \sim U(20,50); \quad (19)$$

$$CE_e \sim U(30,60), \forall e \in \{E:e \neq 0\}; \quad (20)$$

$$CR = 100, CA = 70. \quad (21)$$

3.2 参数设置及校核

在设备和助理资源约束问题中,随着问题规模的变化,设备和助理数量不尽相同,常规设置如表 1 所示。

表 1 设备—助理数量表

I	NA	NE
53	8	12
58	8	12
75	10	14
83	10	14
89	10	14
94	10	14
111	12	16
297	12	16

种群规模、非支配解选择数量、交叉概率和变异概率等参数采用田口实验进行校核,设置参数的常用选择范围:工序交叉概率 LC : $\{0.5, 0.6, 0.7, 0.8, 0.9\}$,设备交叉变异概率 $EC \& EM$: $\{0.3, 0.4, 0.5, 0.6, 0.7\}$,助理交叉变异概率 $AC \& AM$: $\{0.3, 0.4, 0.5, 0.6, 0.7\}$,工序变异概率 LM : $\{0.5, 0.6, 0.7, 0.8, 0.9\}$,种群规模 $M_q + M_z$: $\{40, 45, 50, 55, 60\}$,非支配解选择数量 M_q : $\{15, 20, 25, 30, 35\}$ 。通过 MINITAB 进行参数组合,得到 16 组实验组合,通过算法求取相应的目标函数值,得到的结果如表 2 所示。

表 2 实验组合及结果

	LC	$EC \& EM$	$AC \& AM$	LM	$M_q + M_z$	M_q	Obj
1	0.5	0.3	0.3	0.5	40	15	1 076
2	0.5	0.4	0.4	0.6	45	20	1 055
3	0.5	0.5	0.5	0.7	50	25	917
4	0.5	0.6	0.6	0.8	55	30	1 068
5	0.5	0.7	0.7	0.9	60	35	1 015
6	0.6	0.3	0.4	0.7	55	35	1 012

续表 2								续表 2							
7	0.6	0.4	0.5	0.8	60	15	1 144	18	0.8	0.5	0.3	0.8	45	35	1 160
8	0.6	0.5	0.6	0.9	40	20	1 359	19	0.8	0.6	0.4	0.9	50	15	1 071
9	0.6	0.6	0.7	0.5	45	25	1 005	20	0.8	0.7	0.5	0.5	55	20	1 263
10	0.6	0.7	0.3	0.6	50	30	1 089	21	0.9	0.3	0.7	0.8	50	20	1 037
11	0.7	0.3	0.5	0.9	45	30	986	22	0.9	0.4	0.3	0.9	55	25	1 077
12	0.7	0.4	0.6	0.5	50	35	1 151	23	0.9	0.5	0.4	0.5	60	30	854
13	0.7	0.5	0.7	0.6	55	15	955	24	0.9	0.6	0.5	0.6	40	35	1 296
14	0.7	0.6	0.3	0.7	60	20	881	25	0.9	0.7	0.6	0.7	45	15	885
15	0.7	0.7	0.4	0.8	40	25	1 076								
16	0.8	0.3	0.6	0.6	60	25	1 099								
17	0.8	0.4	0.7	0.7	40	30	1 259								

依据目标函数值的最小化特性,采用田口实验中望小特性的质量损失函数模型对表 2 进行分析,结果如图 5 所示。

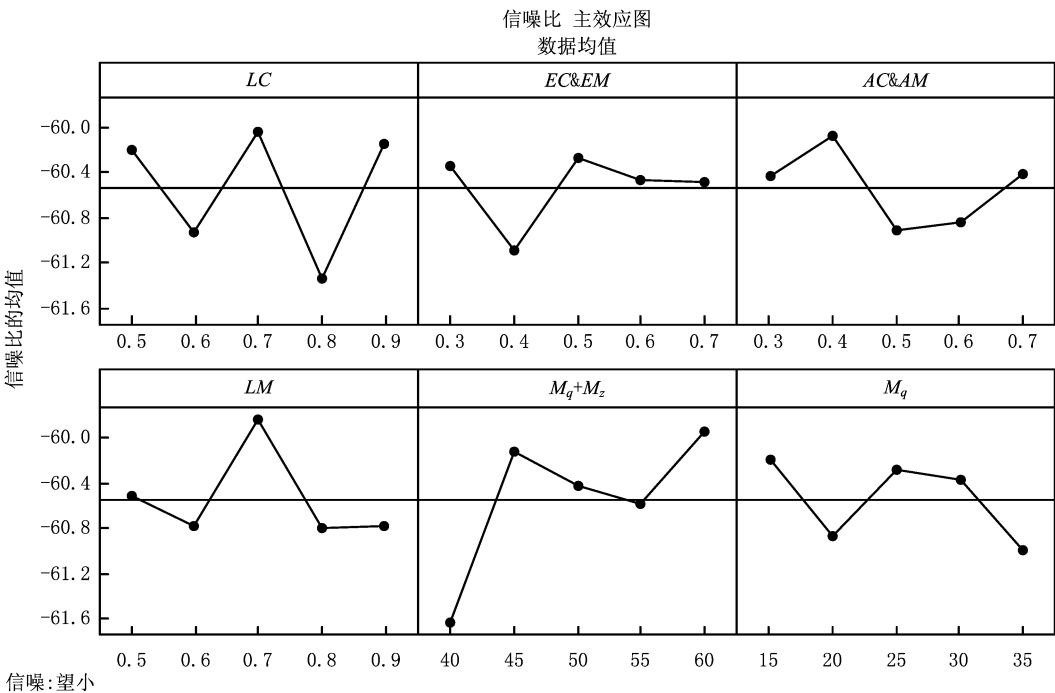


图5 信噪比—主效应图

信噪比是影响产品质量特性的主效应与误差效应的比值,其值越大,稳健性越好。由图 5 可以发现,最佳参数组合方案为 $LC=0.7$, $LM=0.7$, $EC\&EM=0.5$, $AC\&AM=0.4$, $M_q+M_z=60$, $M_q=15$ 。

3.3 算法性能分析

为验证所提算法改进的有效性,通过 C 语言程序对算法进行求解,并与 NSGA-Ⅱ^[16]进行对比,所得实验数据如表 3 所示。

表 3 实验数据对比表

I	CT	NSGA-Ⅱ			BNSGA-Ⅱ			续表 3							
		J	Cost	Efficiency	J	Cost	Efficiency	53	2 338	<u>3</u>	944	0.993 6	<u>3</u>	<u>888</u>	<u>0.997 4</u>
53	2 004	<u>3</u>	1 433	0.851 5	<u>3</u>	<u>1 246</u>	<u>0.99 83</u>	53	2 806	<u>2</u>	916	0.911 1	<u>2</u>	<u>444</u>	<u>0.998 9</u>
								53	3 507	<u>2</u>	947	0.972 4	<u>2</u>	<u>618</u>	<u>0.983 4</u>
								58	86	<u>8</u>	2 474	0.968 3	<u>8</u>	<u>2 024</u>	<u>0.973 6</u>
								58	97	<u>8</u>	2 314	<u>0.992 8</u>	<u>8</u>	<u>1 647</u>	0.986 5
								58	104	8	2 276	0.978 7	<u>7</u>	<u>1 721</u>	<u>0.992 1</u>
								58	111	<u>6</u>	2 378	<u>0.984 8</u>	<u>6</u>	<u>1 581</u>	0.983 4
								75	43	<u>23</u>	3 781	0.896 7	<u>23</u>	<u>3 126</u>	<u>0.921 3</u>
								75	46	<u>22</u>	4 362	<u>0.936 6</u>	<u>22</u>	<u>3 231</u>	0.928 7
								75	50	22	3 810	<u>0.964 3</u>	<u>20</u>	<u>3 035</u>	0.929 8
								75	56	19	4 057	0.936 3	<u>17</u>	<u>2 853</u>	<u>0.946 0</u>
								83	3 786	10	2 947	<u>0.982 8</u>	<u>9</u>	<u>2 165</u>	0.961 6
								83	4 732	<u>7</u>	2 372	<u>0.995 2</u>	<u>7</u>	<u>1 974</u>	0.955 8

续表 3

83	6 883	<u>5</u>	1 909	0.994 4	<u>5</u>	<u>1 530</u>	<u>0.995 2</u>
83	8 412	<u>3</u>	1 732	<u>0.999 1</u>	<u>3</u>	<u>1 352</u>	0.965 0
89	75	14	3 000	<u>0.965 3</u>	<u>13</u>	<u>2 383</u>	0.951 1
89	87	<u>10</u>	2 824	<u>0.965 1</u>	<u>10</u>	<u>2 238</u>	<u>0.965 1</u>
89	103	<u>9</u>	2 736	0.971 5	<u>9</u>	<u>1 717</u>	<u>0.976 7</u>
89	127	<u>7</u>	2 307	0.972 6	<u>7</u>	<u>1 537</u>	<u>0.979 1</u>
94	176	15	3 225	<u>0.983 4</u>	<u>14</u>	<u>2 347</u>	0.953 1
94	192	13	2 939	0.982 1	<u>11</u>	<u>2 090</u>	<u>0.990 1</u>
94	234	10	2 669	0.973 5	<u>8</u>	<u>2 035</u>	<u>0.986 4</u>
94	263	7	2 545	0.982 5	7	1 591	0.982 5
111	5 755	15	3 773	0.972 9	<u>14</u>	<u>3 183</u>	<u>0.973 7</u>
111	6 016	<u>13</u>	3 986	0.972 6	<u>13</u>	<u>3 691</u>	<u>0.974 0</u>
111	6 540	13	3 391	0.974 5	<u>12</u>	<u>3 119</u>	<u>0.988 8</u>
111	7 520	<u>9</u>	3 373	<u>0.988 3</u>	<u>9</u>	<u>2 466</u>	0.980 7
297	2 049	22	4 636	0.971 8	<u>21</u>	<u>3 907</u>	<u>0.987 0</u>
297	2 322	<u>18</u>	4 292	<u>0.987 5</u>	<u>18</u>	<u>3 758</u>	0.982 6
297	2 488	16	4 021	0.989 1	<u>15</u>	<u>3 505</u>	<u>0.992 3</u>
297	2 787	15	3 618	0.990 5	<u>14</u>	<u>3 163</u>	<u>0.991 6</u>

表 3 中, NSGA-II 为改进前的算法, 为适应多重资源约束问题, 在编码解码阶段采用本文方法。通过对比, 发现所提算法在不同节拍下, 得到的工位少于 NSGA-II, 且成本低、效率高, 表明所提算法的性能优于 NSGA-II。主要由于所提算法采用非回溯的 Pareto 层级构造, 并提出基于概率的层次化遗传操作。该操作扩充了邻域结构, 增强了算法的寻优能力, 能够避免算法陷入局部优化。

为综合评价多目标优化算法的性能, 采用非支配解比率 RP 、Pareto 前沿解收敛性 CP 和个体间距度量指标 SP ^[17] 3 个指标分析 BNSGA-II、多目标遗传算法 (Multi-Objective Genetic Algorithm, MOGA) 和非支配排序遗传算法 (Non-dominated

Sorting Genetic Algorithm, NSGA)。对比结果如表 4 所示, 并在图 6 绘制 3 种算法求解典型案例得到的 Pareto 解集, 从空间分析 3 种算法的分布性。

(1) 非支配解比率 RP 对于每一个 Pareto 解集 $S_i (i=1, 2, 3, \dots, s, S=S_1+S_2+\dots+S_i+\dots+S_s)$, 非支配解比率为解集 S_i 中不被其他解集支配的解所占的比重, 其值越高, 解集 S_i 的性能越好。

$$RP(S_i) = \frac{|S_i - \{x \in S_i \mid \exists y \in S: y \succ x\}|}{|S_i|} \quad (22)$$

式中: $y \succ x$ 表示 y 支配 x ; $|S_i|$ 表示解集 S_i 的数量。

(2) Pareto 前沿解收敛性 CP 表示 Pareto 解集 S_i 与近似 true-Pareto 解集 TP 的不同。

$$CP(S_i) = \frac{\sum_{j=1}^{|S_i|} dt_j}{|S_i|}; \quad (23)$$

$$dt_j = \min_{n=1}^{|TP|} \sqrt{\sum_{m=1}^2 \left[\frac{f_m(x_j) - f_m(x_n)}{f_m^{\max} - f_m^{\min}} \right]^2} \quad (24)$$

式中: TP 为目前所有算法所求得的非支配解的集合; dt_j 为集合 S_i 中的第 j 个解与解集 TP 中非支配解的最小欧式距离; f_m^{\max} 和 f_m^{\min} 为第 m 个目标函数的最大和最小值。

(3) 个体间距度量指标 SP 用于衡量 Pareto 集合的分布性。

$$SP(S_i) = \frac{\sum_{j=1}^{|S_i|-1} \|sd_j - \overline{sd}\|}{sd_j + (|S_i| - 1) \overline{sd}}; \quad (25)$$

$$sd_j = \sqrt{\sum_{m=1}^2 \left[\frac{f_m(x_j) - f_m(x_{j+1})}{f_m^{\max} - f_m^{\min}} \right]^2} \quad (26)$$

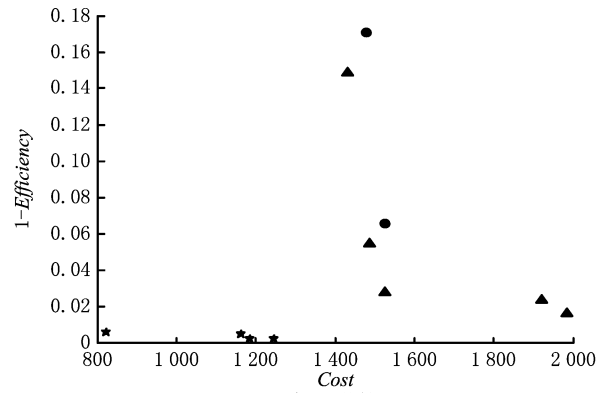
式中: sd_j 为 Pareto 集合中相邻解之间的欧氏距离; \overline{sd} 为 sd_j 的均值。

表 4 多目标优化算法对比结果

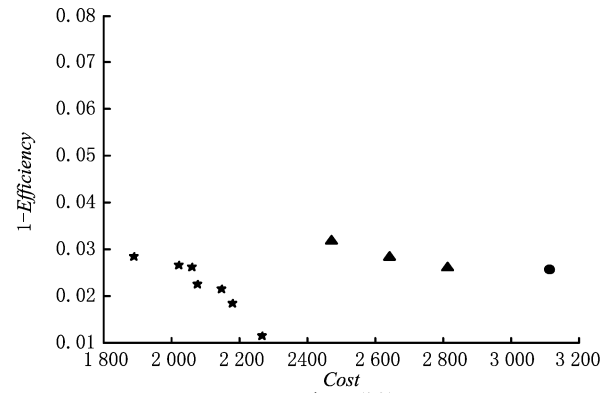
I	CT	MOGA			NSGA			BNSGA-II		
		RP	CP	SP	RP	CP	SP	RP	CP	SP
53	2 004	0	0.256	0	0	0.379	0.685	<u>1</u>	<u>0</u>	<u>0.821</u>
53	2 338	0	0.284	—	0	0.277	0	<u>1</u>	<u>0</u>	<u>0.377</u>
53	2 806	0	0.246	0	0	0.208	0.229	<u>1</u>	<u>0</u>	<u>0.925</u>
53	3 507	0	0.261	0	0	0.033	0.405	<u>1</u>	<u>0</u>	<u>0.735</u>
58	86	0	0.664	0	0	0.307	0	<u>1</u>	<u>0</u>	<u>0.438</u>
58	97	0	0.153	—	<u>1</u>	0.001	<u>0.800</u>	<u>1</u>	<u>0</u>	0
58	104	0	0.502	<u>0</u>	0	0.614	<u>0</u>	<u>1</u>	<u>0</u>	<u>0</u>
58	111	0	0.153	<u>0.423</u>	0	0.303	0.407	<u>1</u>	<u>0</u>	0.401
75	43	0	0.225	0.434	0	0.086	0.277	<u>1</u>	<u>0</u>	<u>0.450</u>

续表 4

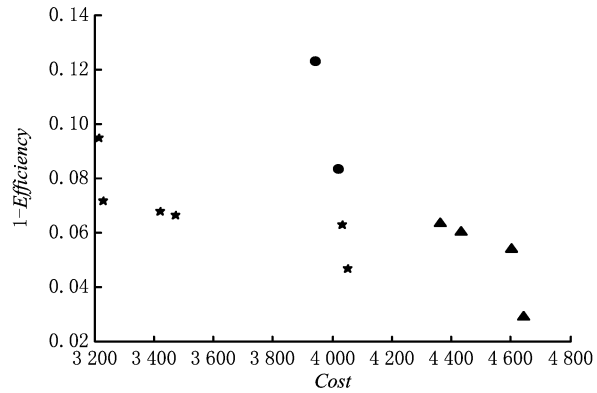
75	46	0	0.093	0.263	0.250	0.097	0.415	<u>1</u>	<u>0</u>	<u>0.902</u>
75	50	0	0.231	0.219	<u>1</u>	0.001	0.196	<u>1</u>	<u>0</u>	<u>0.638</u>
75	56	0	0.676	0.097	0	0.834	0.031	<u>1</u>	<u>0</u>	<u>0.719</u>
83	3 786	0	0.226	0.078	<u>1</u>	0.032	<u>1.265</u>	0.833	<u>0.014</u>	0.543
83	4 732	0	0.452	<u>0.984</u>	<u>1</u>	<u>0.003</u>	0.333	0.750	0.093	0.618
83	6 883	0	0.183	0.421	0.666	0.001	0.006	<u>1</u>	<u>0</u>	<u>0.436</u>
83	8 412	0	0.08	0.070	<u>1</u>	0.043	0	0.444	<u>0.003</u>	<u>0.671</u>
89	75	0	0.095	0.337	0.600	0.029	0	<u>0.833</u>	<u>0.023</u>	<u>0.736</u>
89	87	0	0.157	0.097	<u>0.800</u>	0.037	0	0.750	<u>0.001</u>	<u>0.808</u>
89	103	0	0.900	0	0	0.888	0.400	<u>1</u>	<u>0</u>	<u>0.424</u>
89	127	0.5	0.025	0	0.333	0.092	0.171	<u>0.750</u>	<u>0.014</u>	<u>0.458</u>
94	176	0	0.138	0.021	<u>1</u>	<u>0</u>	0	<u>1</u>	<u>0</u>	<u>0.621</u>
94	192	0	0.337	0.024	<u>1</u>	0	0.250	<u>1</u>	<u>0</u>	<u>0.392</u>
94	234	0	0.472	<u>0.435</u>	0	0.659	0.350	<u>1</u>	<u>0</u>	0.103
94	263	0	0.131	0	0	0.102	0	<u>1</u>	<u>0</u>	<u>0.664</u>
111	5 755	0	0.299	0.494	0.200	0.001	<u>0.560</u>	<u>1</u>	<u>0</u>	0.429
111	6 016	0	0.103	0	0	0.244	0.363	<u>1</u>	<u>0</u>	<u>0.642</u>
111	6 540	0	0.399	0	0	0.234	0.518	<u>1</u>	<u>0</u>	<u>0.729</u>
111	7 520	0	0.117	0	<u>1</u>	<u>0</u>	<u>0.913</u>	<u>1</u>	<u>0</u>	0
297	2 049	0	0.525	0	0	0.547	0.222	<u>1</u>	<u>0</u>	<u>0.627</u>
297	2 322	0	0.107	0	0	0.164	0	<u>1</u>	<u>0</u>	<u>0.861</u>
297	2 488	0	0.377	0	0	0.410	<u>0.571</u>	<u>1</u>	<u>0</u>	0.472
297	2 787	0	0.490	0	0	0.424	0.142	<u>1</u>	<u>0</u>	<u>0.550</u>



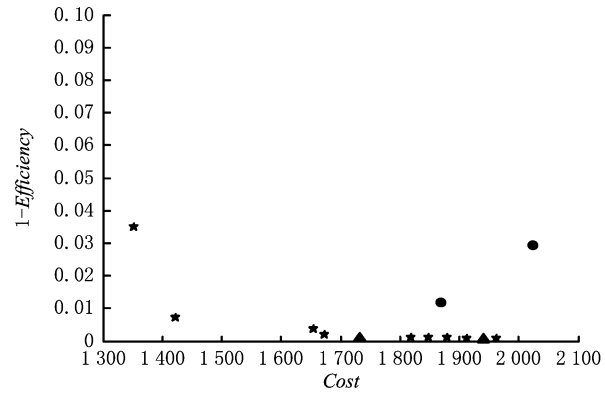
a 工序53, 节拍2 004



b 工序58, 节拍86



c 工序75, 节拍46



d 工序83, 节拍8 412

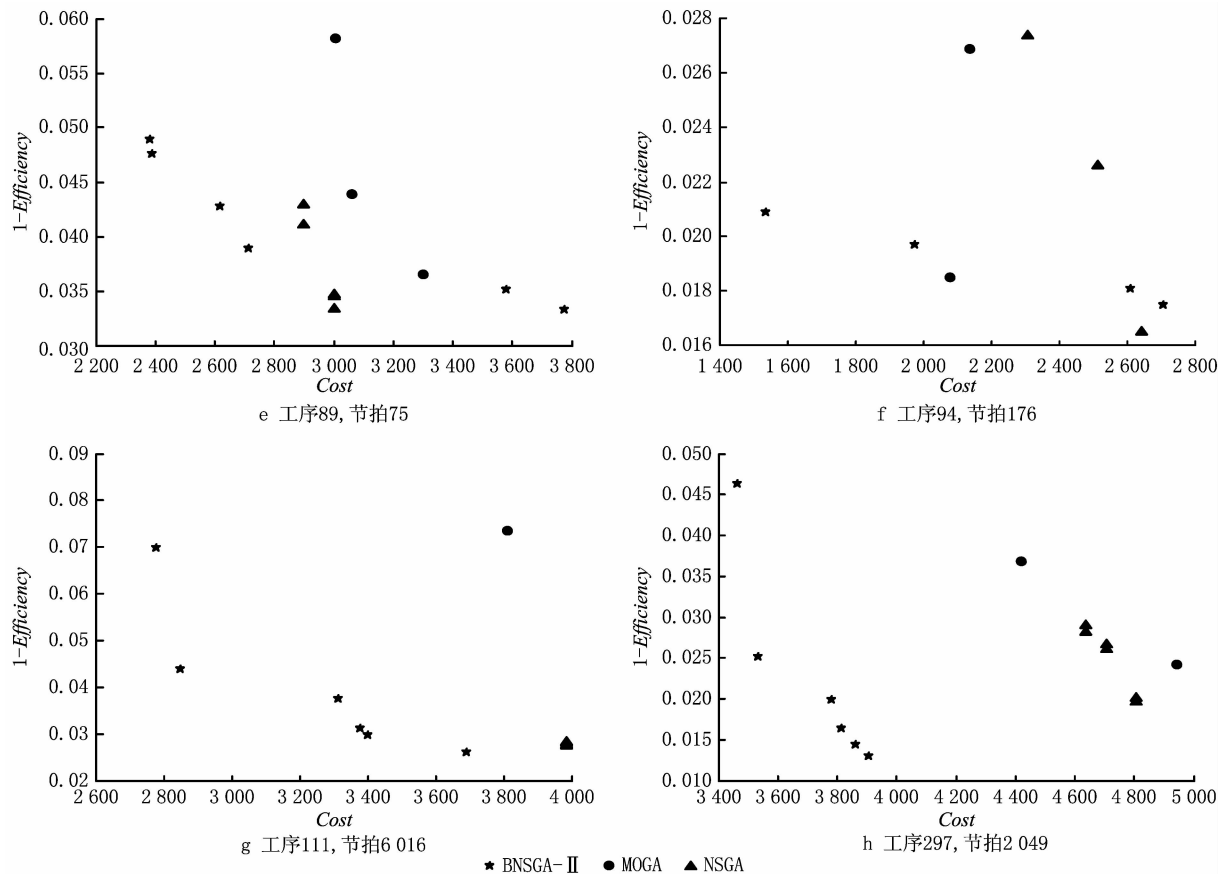


图6 Pareto前沿解对比图

表 4 中, BNSGA-II 算法的 RP 值基本为 1, 表明该算法得到的 Pareto 解能支配 MOGA 和 NSGA 算法的 Pareto 集; 上述案例中, 近似 true-Pareto 解集 TP 为 3 个算法综合得到的最优 Pareto 集, 而 BNSGA-II 算法的 CP 值均最小, 表明 BNSGA-II 算法的收敛性优于 MOGA 和 NSGA 算法; 同时, BNSGA-II 算法的 SP 值基本大于另外两种算法, 表明 BNSGA-II 算法在各种案例中都具有良好的分布性。

4 结束语

为保证 U 型装配线的生产率, 提高装配线的稳定性, 本文综合考虑设备和助理分配问题, 构建了基于资源分配的成本和效率双目标的 U 型装配线平衡问题模型。针对工序、设备和助理分配等多决策变量, 采用 Benders 分解法, 将问题分解为分配设备助理的主问题和分配工序的子问题, 降低了模型求解的复杂度。提出基于 Benders 分解法的 BNSGA-II 实现模型的求解。通过标杆案例, 分析 BNSGA-II, MOGA 和 NSGA 多目标算法的非支配解比率、

Pareto 前沿解收敛性和个体间距离度量指标, 证明了所提算法的优越性。由于在装配线中考虑资源分配的研究较少, 进一步工作可在多重资源约束的 U 型装配线问题中融入区域约束和加强约束, 构建模型并进行求解, 最终得出有效的解决方案。

参考文献:

- [1] MILTENBURG. The U-line balancing problem[J]. Manage Sci, 1994, 40(10): 1378-1388.
- [2] URBAN T L. Note. optimal balancing of U-shaped assembly lines[J]. Management Science, 1998, 44(5): 738-741.
- [3] SCHOLL A, KIEIN R. ULINO; optimally balancing U-shaped JIT assembly lines[J]. International Journal of Product Research, 1999, 37(4): 721-736.
- [4] EREL E, SABUNCUOGLU I, AKSU B A. Balancing of U-type assembly systems using simulated annealing[J]. International Journal of Product Research, 2001, 39(13): 3003-3015.
- [5] GOKCEN H, AGPAK 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.
- [6] HWANG R K, KATAYAMA H, GEN M. U-shaped assembly line balancing problem with genetic algorithm[J]. International Journal of Product Research, 2008, 46(16): 4637-4649.

- [7] SABUNCUOGLU I, EREL E, ALP A. Ant colony optimization for the single model U-type assembly line balancing problem[J]. International Journal of Product Economics, 2009, 120(2): 287-300.
- [8] ZHANG Zikai, TANG Qiuhua, ZHANG Liping, et al. Balancing large-size mixed-model U-type assembly lines via improved genetic algorithm[J]. Machinery Design & Manufacture, 2016(1): 137-140 (in Chinese). [张子凯, 唐秋华, 张利平, 等. 改进遗传算法求解大规模混流 U 型装配线问题[J]. 机械设计与制造, 2016(1): 137-140.]
- [9] MANAVIZADEH N, HOSSEINI N S, RABBANI M, et al. A simulated annealing algorithm for a mixed model assembly U-line balancing type-I problem considering human efficiency and just-in-time approach[J]. Computers & Industrial Engineering, 2013, 64(2): 669-685.
- [10] KARA Y, OZGUVEN C, YALCIN N, et al. Balancing straight and U-shaped assembly lines with resource dependent task times[J]. International Journal of Product Research, 2011, 49(21): 6387-6405.
- [11] 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.
- [12] KARA K, OZCAN U, PEKER A. Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives[J]. Applied Mathematics and Computation, 2007, 184(2): 566-588.
- [13] 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.
- [14] TANG Qiuhua, LIN Bin, HE Xiaoxia, et al. Balancing U-shaped assembly lines via stochastic chance-constrained programming[J]. Computer Integrated Manufacturing Systems, 2015, 22(4): 955-964 (in Chinese). [唐秋华, 林斌, 何晓霞, 等. 基于随机机会约束规划的 U 型装配线平衡优化[J]. 计算机集成制造系统, 2015, 22(4): 955-964.]
- [15] TANG Qiuhua, HU Jin, ZHANG Liping, et al. A hybrid multi-objective hybrid genetic algorithm for automobile production sequencing problems[J]. China Mechanical Engineering, 2015, 26(16): 2170-2178 (in Chinese). [唐秋华, 胡进, 张利平, 等. 面向汽车投产排序的混合多目标网格遗传算法[J]. 中国机械工程, 2015, 26(16): 2170-2178.]
- [16] 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): 182-197.
- [17] LI Zixiang, TANG Qiuhua, 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, 2016, 135: 508-522.

作者简介:

张子凯(1994—),男,湖北襄阳人,硕士研究生,研究方向:生产过程控制、智能算法, E-mail: zhangzikai0703@126.com;

唐秋华(1970—),女,湖北利川人,土家族,教授,博士,博士生导师,研究方向:现代制造系统、制造业信息化、工业工程与管理,通信作者, E-mail: tangqiuhua@wust.edu.cn;

张利平(1983—),女,湖北武汉人,副教授,研究方向:智能算法、先进制造技术。