

任务分区及工位约束下装配线第二类平衡研究

徐立云, 蔡炳杰, 曲宝升, 李爱平

(同济大学 现代制造技术研究所, 上海 201804)

摘要: 装配线平衡直接影响产品产量, 也是在装配线布局初始或重构时必须面临的问题. 针对第二类装配线平衡问题(ALBP-2), 构建了一种考虑优先关系约束、任务分区约束和工位约束的多目标优化模型. 为提高模型求解效率, 采用逐步缩小节拍搜索范围的动态步长方法, 提出基于自动机回溯算法的改进粒子群算法, 提高了任务分配合理性, 快速搜索出具有最小节拍和负载平滑系数的任务分配方案. 引入任务分配矩阵表示每个工位上的任务分配情况, 使结果描述更加明确. 通过案例分析验证了所提模型和方法的有效性.

关键词: 第二类装配线平衡问题; 粒子群算法; 多目标优化; 任务分区约束; 工位约束

中图分类号: TB491

文献标志码: A

Assembly Line Balancing Problem-2 with Task Zoning Constraints and Workstation Related Constraints

XU Liyun, CAI Bingjie, QU Baosheng, LI Aiping

(Institute of Advanced Manufacturing Technology, Tongji University, Shanghai 201804, China)

Abstract: The balance of assembly line has a great effect on the output of products, and is also a problem to be dealt with at the phases of designing and reconfiguring. Aimed at the current assembly line balancing problem type-2 (ALBP-2), a multi-objective model was proposed, taking task sequence constraints, task zoning constraints and workstation related constraints into account, to minimize cycle time and work-loading smoothness index. First, a dynamic step length method was applied, which gradually shrank the searching span of cycle time, to improve the efficiency of computing. Then, an improved PSO algorithm based on a Back Algorithm (BA) was introduced to promote the rationality of task-allocation. After that task allocation matrix was described to show details of task allocations in stations and clarify the result. Finally, a case was illustrated to prove the

effectiveness of the proposed method.

Key words: assembly line balancing problem-2 (ALBP-2); PSO algorithm; multi-objective optimization; task zoning constraints; workstation related constraints

装配是指将若干零部件结合成产品的过程, 是汽车、电器等制造行业生产产品的核心环节. 规模生产中常见的装配线是将装配过程分解成若干个任务, 并将这些任务分配到不同的工位上执行的一种表现形式. 装配线平衡问题(ALBP)是装配线设计布局的重要问题之一. 根据优化目标不同, 装配线平衡问题可以分为两类: 在生产节拍与加工任务已知下, 求最小工位数量的第一类装配线平衡问题(ALBP-1); 在工位数与加工任务已知下, 求最小节拍的第二类装配线平衡问题(ALBP-2)^[1]. 针对 ALBP-2, 国内外很多学者进行了深入的研究, 主要研究方向为算法改进与模型优化. 对于算法改进, 大量学者将蚁群算法、粒子群算法、混合粒子群算法等智能算法及规则算法应用到了 ALBP-2 上^[1-5]. 在模型优化方面, 周亮^[6]通过减少一组约束以改进 ALBP-2 数学模型. Hamta 等^[7]研究了涉及最小节拍、最小设备成本及最小平滑系数的多目标第二类装配线平衡问题. Saif 等^[8]解决了任务操作时间不确定的第二类装配线平衡问题. 吴尔飞^[9]提出了双边装配线第二类平衡问题的数学模型, 在模型中不仅考虑单边装配线存在的优先关系约束, 而且还考虑任务操作方位以及双边并行作业的要求. 钱雄文^[10]提出了多人共站装配线第二类平衡问题, 并通过实例验证了算法的可行性.

文献[6-10]针对 ALBP-2 所构建的数学优化模型很好地解决了相应的案例问题, 但是对于更加普遍的问题, 模型则没有很好地表示工位和任务的约

收稿日期: 2015-01-17

基金项目: 国家科技重大专项(2013ZX04012071)

第一作者: 徐立云(1973—), 男, 教授, 博士生导师, 工学博士, 主要研究方向为智能制造、系统建模与优化、产品数字化设计与管理.

E-mail: Lyxu@tongji.edu.cn

束. 并且由于约束的表示又是建模过程中的重难点, 因此对 ALBP-2 中的约束问题进行研究有重要意义. 为深入探讨 ALBP-2 的约束问题, 本文以单品种单边装配线平衡为问题背景, 构建多约束、多目标优化模型, 采用逐步缩小节拍搜索范围的动态步长方法及基于自动机回溯算法的改进粒子群算法, 使装配线达到平衡, 且得到更加符合实际情况的任务分配结果. 在结果描述方面, 采用任务分配矩阵以表示每个工位上的任务分配情况. 最后, 通过实例分析, 验证了模型及方法的有效性.

1 问题描述及数学模型

1.1 问题描述及相关假设

单品种单边装配线是最普遍的装配线之一, 该类装配线除了考虑典型的优先关系约束外, 还会存在如下约束: ① 任务分区约束 (task zoning constraints), 出于设备、工艺与安全性考虑, 部分工位只能接受特定的一些任务; ② 工位约束 (workstation related constraints), 某些任务由于任务性质只能分配到特定工位. 针对该类装配线, 以往研究在数学模型构建过程中往往忽略实际工作过程中的部分约束, 如文献[10]只考虑到优先关系约束, 使得结果与实际情况有一定差距.

本文在同时考虑优先关系约束、任务分区约束及工位约束的情况下, 建立 ALBP-2 优化模型. 假设: ①每个任务都有稳定的任务时间; ②必须将所有任务都分配到工位中; ③工位个数是确定的; ④每个任务都已经是最小任务单元, 不能划分; ⑤任务的分配不能违反优先关系约束、任务分区约束及工位约束. 因此, 已知一个存在优先关系约束、任务分区约束及工位约束的任务集, 任务个数为 n , 工位数为 m , 在满足约束的前提下, 分配各任务使装配线的节拍最小并使各工位的负荷尽可能相等.

1.2 数学模型的建立

1.2.1 变量定义

本文主要变量及其含义定义如下: I 为任务矩阵, $I=(1, 2, \dots, n)$ 为行向量, 由所有任务序号构成; I_1, I_2 为子任务矩阵, I_1, I_2 均由 I 部分元素构成; $t(i)$ 为任务 i 的操作时间, i 为 I 内元素; P 为优先关系矩阵, $P=(P_{i_1, i_2})_{n \times n}$, 如果任务 i_1 是任务 i_2 的优先任务, $P_{i_1, i_2}=1$, 否则为 0; J 为工位矩阵, $J=(1, 2, \dots, m)$, 由所有 I 位序号构成; C_T 为装配线的节拍;

B 为装配线的平衡率; $S_T(j)$ 为工位 j 的装配时间, j 为 J 内元素; n 为任务数; m 为工位数.

1.2.2 任务分配矩阵

为了描述任务在各工位的分配情况, 引入任务分配矩阵

$$S_{eq} = \begin{pmatrix} S_{eq}(1,1) & S_{eq}(1,2) & \cdots & S_{eq}(1,n) \\ S_{eq}(2,1) & S_{eq}(2,2) & \cdots & S_{eq}(2,n) \\ \vdots & \vdots & & \vdots \\ S_{eq}(m,1) & S_{eq}(m,2) & \cdots & S_{eq}(m,n) \end{pmatrix} \quad (1)$$

式中: S_{eq} 的每一行非零元素代表该工位上分配到的任务序号. 该矩阵有如下特点: ①行最简且行满秩矩阵; ②非零元素值为 1 到 n , 且彼此不重复. 其中①, ②分别保证了所有工位分配到了任务以及任务的唯一性.

图 1 为某 S_{eq} 矩阵示意图, 代表含义如表 1 所示.

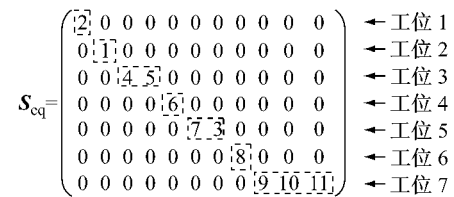


图 1 任务分配矩阵示意图

Fig.1 Task allocation matrix

表 1 任务在工位上的分配结果

Tab.1 Allocation results of tasks in stations							
工位序号	1	2	3	4	5	6	7
任务序号	{2}	{1}	{4,5}	{6}	{7,3}	{8}	{9,10,11}

1.2.3 任务分配时间矩阵

为了描述各工位用时, 定义任务分配时间矩阵为

$$T_{S_{eq}} = \begin{pmatrix} T_{S_{eq}}(1,1) & T_{S_{eq}}(1,2) & \cdots & T_{S_{eq}}(1,n) \\ T_{S_{eq}}(2,1) & T_{S_{eq}}(2,2) & \cdots & T_{S_{eq}}(2,n) \\ \vdots & \vdots & & \vdots \\ T_{S_{eq}}(m,1) & T_{S_{eq}}(m,2) & \cdots & T_{S_{eq}}(m,n) \end{pmatrix} \quad (2)$$

矩阵中的元素满足

$$T_{S_{eq}}(j,i) = \begin{cases} t(S_{eq}(j,i)), & j = 1, 2, \dots, m, \\ & i = 1, 2, \dots, n, S_{eq}(j,i) \neq 0 \\ 0, & j = 1, 2, \dots, m, \\ & i = 1, 2, \dots, n, S_{eq}(j,i) = 0 \end{cases} \quad (3)$$

则第 j 号工位用时

$$S_T(j) = \sum_{i=1}^n T_{S_{eq}}(j, i), j = 1, 2, \dots, m \quad (4)$$

装配线节拍为各工位装配的最长用时

$$C_T = \max\{S_T(1), S_T(2), \dots, S_T(m)\} \quad (5)$$

1.2.4 ALBP-2 数学模型

决策变量: x_{ij} , 0-1 变量, 当任务元素 i 被分配至第 j 工位时, $x_{ij} = 1$, 否则 $x_{ij} = 0$.

$$\text{约束条件: } \sum_{j=1}^m x_{ij} = 1, i = 1, 2, \dots, n \quad (6)$$

$$S_T(j) \leq C_T, j = 1, 2, \dots, m \quad (7)$$

$$\sum_{j=1}^m (jx_{i_2j} - jx_{i_1j}) \geq 0, \forall (i_1, i_2) \in P \quad (8)$$

$$F_{\text{length}}(F_{\text{intersect}}(S_{eq}(j_1, :), I_1)) + F_{\text{length}}(F_{\text{intersect}}(S_{eq}(j_2, :), I_1)) + \dots + F_{\text{length}}(F_{\text{intersect}}(S_{eq}(j_l, :), I_1)) = F_{\text{length}}(I_1), l \leq m \quad (9)$$

$$F_{\text{length}}(F_{\text{intersect}}(S_{eq}(j, :), I_2)) = F_{\text{length}}(F_{\text{intersect}}(S_{eq}(j, :), I)) \quad (10)$$

假设有 \mathbf{A}, \mathbf{B} 行向量, 那么 $F_{\text{intersect}}(\mathbf{A}, \mathbf{B})$ 表示行向量 \mathbf{A}, \mathbf{B} 的交集元素所组成的行向量; $F_{\text{length}}(\mathbf{A})$ 表示行向量 \mathbf{A} 的维数; $S_{eq}(j, :)$ 表示由 S_{eq} 中第 j 行元素构成的行向量.

单品种单边装配线中, 较小的节拍意味着更高的生产效率, 较小的负载平滑系数意味着更少的设备闲置时间. 构建目标函数如下:

主要目标

$$Z_1 = \min C_T \quad (11)$$

次要目标

$$Z_2 = \min \sqrt{\sum_{j=1}^m (C_T - S_T(j))^2} \quad (12)$$

式(6)保证同一任务只能分配到一个工位上; 式(7)保证每个工位用时不超过节拍; 式(8)保证不能违反任务优先关系; 式(9)是工位约束, 保证子任务集 I_1 中的每个任务必须分配到工位 j_1 或 j_2, \dots , 或 j_l ; 式(10)是任务分区约束, 保证工位 j 上仅能分配到子任务集 I_2 中的任务.

2 求解思路与算法设计

2.1 求解思路

针对上文中所构建的模型, 论文基于动态步长及改进粒子群算法进行问题求解. 采用动态步长方法是为了提高算法效率. 通过粒子群算法, 虽然能够

产生不同的分配方案, 但这些方案本身却有很大的局限性: 任务只是单向地流向工位, 各个工位间无法实现任务的相互流通. 因此, 基于自动机回溯算法对各个方案进行回溯调整, 尽可能使各工位负载趋于相同. 具体步骤如下: ①定界——由已知条件确定最优节拍的大致所在范围, 同时指定初步步长. ②分组——根据步长将该范围区间进行分段, 得到几个区间端点节拍. ③基于自动机回溯算法的改进粒子群算法计算——从小到大逐个取步骤②中得到的节拍, 分别进行以负载平滑系数为适应度的改进 PSO 算法计算, 得到指定节拍下的最优负载平滑系数及所对应的任务分配矩阵. ④当前组最优保留——节拍最小是主要优化目标, 但节拍过小将无法获得合理的任务分配矩阵. 所以, 当节拍从小到大搜索时, 首次得到合理的任务分配矩阵对应的节拍就是当前组的最优节拍, 进而计算得到的最优负载平滑系数与对应任务分配矩阵为该组最佳. ⑤缩小搜索空间——以步骤④中得到的最优节拍值为区间中点, 以现有步长为区间半径, 确定新的最优节拍所在区间. 然后缩短步长, 进入步骤②开始新一组的搜索. ⑥优选及结果输出——循环达到终止步长时, 取节拍最小的那一组为最优结果. 同时输出该节拍下计算得到的任务分配矩阵.

2.2 算法设计

2.2.1 动态步长算法

本文在求解过程中, 采用逐步缩小节拍搜索范围的动态步长方法来提高整体算法效率, 原理如图 2 所示, 具体表现如下:

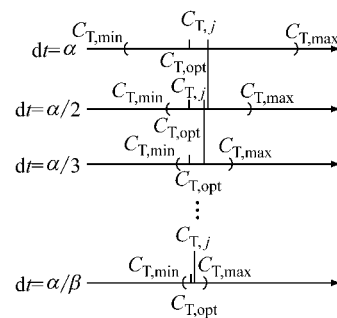


图 2 动态步长原理

Fig.2 Theory of dynamic step

(1) 定界 确定节拍变动范围. $C_{T,\min}$ 由式(13)求得. $C_{T,\max}$ 取带有绝对冗余量的设定节拍.

$$C_{T,\min} = \max\left\{\frac{\sum_{i=1}^n t(i)}{m}, \max(t(i))\right\} \quad (13)$$

(2) 参数初始化 动态步长方法中有 3 个较为重要的参数 α, β, dt , 它们分别表示初始步长, 控制精度因子及当前步长.

(3) 进入计算主体

$$\begin{cases} C_{T,1} = C_{T,\min} \\ C_{T,2} = C_{T,1} + dt \\ \vdots \\ C_{T,l} = C_{T,l-1} + dt \\ \vdots \\ C_{T, \lfloor \frac{C_{T,\max} - C_{T,\min}}{dt} \rfloor + 1} = C_{T, \lfloor \frac{C_{T,\max} - C_{T,\min}}{dt} \rfloor} + dt \end{cases} \quad (14)$$

$$\begin{cases} C_{T,1} \Rightarrow \Delta_1, \mathbf{S}_{\text{eq},1} \\ C_{T,2} \Rightarrow \Delta_2, \mathbf{S}_{\text{eq},2} \\ \vdots \\ C_{T,l} \Rightarrow \Delta_l, \mathbf{S}_{\text{eq},l} \\ \vdots \\ C_{T, \lfloor \frac{C_{T,\max} - C_{T,\min}}{dt} \rfloor + 1} \Rightarrow \Delta_{\lfloor \frac{C_{T,\max} - C_{T,\min}}{dt} \rfloor + 1}, \mathbf{S}_{\text{eq}, \lfloor \frac{C_{T,\max} - C_{T,\min}}{dt} \rfloor + 1} \end{cases} \quad (15)$$

$$\Delta_l = \min \sqrt{\sum_{j=1}^m (C_{T,l} - S_T(j))^2} \quad (16)$$

式(15)中, 针对 $C_{T,l}$ 应用 1.2.4 中提出的模型求解 Δ_l 及 $\mathbf{S}_{\text{eq},l}$ 的过程是利用改进 PSO 算法进行求解的环节. $\Delta_l, \mathbf{S}_{\text{eq},l}$ 分别为 $C_{T,l}$ 对应的最优目标函数值与最优任务分配矩阵. C_T 从小到大搜索一直达到 $C_{T,j}$ 时, 在该次搜索内首次产生合理任务分配矩阵, 从而完成该种群的改进 PSO 算法运算, 得到该种群最佳的任务分配矩阵 $\mathbf{S}_{\text{eq},j}$ 与对应的适应度函数值 Δ_j .

$$\begin{cases} C_{T,\max} = C_{T,j} + dt \\ C_{T,\min} = C_{T,j} - dt \end{cases} \quad (17)$$

依据式(17)计算后缩小了最优节拍所在范围, 然后直接结束该组的搜索.

(4) 缩短步长 令 $dt = \alpha/2$, 重复步骤(3), 得到第二组粒子群的各项最佳值: $C_{T,j}, \Delta_j, \mathbf{S}_{\text{eq},j}$ 以及一个更精确 $C_{T,\text{opt}}$ 所在的范围 $[C_{T,\min}, C_{T,\max}]$.

(5) 循环 循环至终止步长 $dt = \alpha/\beta$, 得到该组的 $C_{T,j}, \Delta_j, \mathbf{S}_{\text{eq},j}$. 输出最小节拍 $C_{T,\text{opt}} = C_{T,j}$ (由所有组中最小的 $C_{T,j}$ 来近似 $C_{T,\text{opt}}$) 对应分配方案为 $\mathbf{S}_{\text{eq},\text{opt}} = \mathbf{S}_{\text{eq},j}$.

(6) 修正 动态步长取值是以非整数变化, 故通过步骤(5)得到的 $C_{T,\text{opt}}$ 理论上会比真实值大, 需要进行修正. 可采用步骤(5)中最后得到的任务分配矩阵 $\mathbf{S}_{\text{eq},\text{opt}}$ 来修正 $C_{T,\text{opt}}$, 修正节拍为

$$C'_{T,\text{opt}} = \min\{C_{T,\text{opt}}, \max\{S_T(j)\}\}, \quad j = 1, 2, \dots, m \quad (18)$$

2.2.2 改进 PSO 算法

改进 PSO 算法取负载平滑系数为适应度函数, 具体应用过程为: 在 D 维搜索空间中, 随机生成一群粒子, 位置及速度分别为 $X^{(k)} = (X_1^{(k)}, X_2^{(k)}, \dots, X_{\text{popsize}}^{(k)})$, $V^{(k)} = (V_1^{(k)}, V_2^{(k)}, \dots, V_{\text{popsize}}^{(k)})$, k 表示迭代次数, 此时 $k=1$. 第 i 个粒子表示为 $X_i^{(k)} = (x_{i1}, x_{i2}, \dots, x_{iD})$, $V_i^{(k)} = (v_{i1}, v_{i2}, \dots, v_{iD})$. 任务排序矩阵 \mathbf{b} 为 $X_i^{(k)}$ 中元素按从小到大排序的序号行向量. 定义 \mathbf{b} 中的元素值为对应任务序号. 逐个粒子判断任务排序矩阵中元素是否满足优先关系矩阵 \mathbf{P} . 若不满足, 则按优先关系进行元素的对调, 直到满足为止^[11]. 然后, 对任务排序矩阵 \mathbf{b} 进行分配, 依据节拍条件约束(7)完成任务的分配, 初步生成任务分配矩阵. 接着, 基于自动机回溯算法进行任务流动, 检查生成的任务分配矩阵是否满足约束(9)和(10), 若满足则进行适应度的计算, 之后与已有 $p_{\text{best},i}$ 与 g_{best} 比较, 判断是否更新 $p_{\text{best},i}$ 与 g_{best} , 否则直接进入下一轮回溯算法的运行. 该环节的终止条件是达到回溯循环上限, 达到上限后更新粒子. 更新后粒子的速度与位置为

$$v_{id}^{(k+1)} = wv_{id}^{(k)} + c_1 r_1 (P_{id} - x_{id}^{(k)}) + c_2 r_2 (P_{gd} - x_{id}^{(k)}) \quad (19)$$

$$x_{id}^{(k+1)} = x_{id}^{(k)} + v_{id}^{(k+1)}, \quad i = 1, 2, \dots, n, \quad d = 1, 2, \dots, D \quad (20)$$

式中: w 为惯性权重; c_1, c_2 为学习因子; r_1, r_2 为 $[0, 1]$ 之间的随机数; $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$, 为个体极值; $P_g = (P_{g1}, P_{g2}, \dots, P_{gD})$, 为全局极值. 之后进行迭代, 循环至 $k = M_{\text{axgen}}$ (循环代数上限), 最后得到该种群下最优目标函数值 Δ_l ($\Delta_l = g_{\text{best}}$) 与最优任务分配矩阵 $\mathbf{S}_{\text{eq},l}$.

2.2.3 初始任务分配矩阵的构建

在完成对调后的任务排序矩阵 \mathbf{b} 基础上生成任务分配矩阵, 即进行任务的分配. 该部分算法的核心思想是对任务排序矩阵 \mathbf{b} 按指定节拍分段, 然后逐段替代原始为零矩阵的任务分配矩阵中的元素. 这个过程的目标是生成满足优先关系约束且任务在指定节拍下能合理分配——不出现最后一个工位上没任务或最后一个工位上按节拍要求分配后还有多余任务——的任务分配矩阵, 对于不合格的粒子直接剔除. 图 3 为一个分配案例.

2.2.4 自动机回溯算法

自动机回溯算法主要功能是对相邻任务在相邻工位间作尝试性流动, 若能达到流动后的任务分配方案满足约束(9)和(10), 同时具有更好的负载平滑系数, 则予以保留生成新的任务分配矩阵. 否则, 将

任务分配矩阵恢复到任务流动前的形式^[12]. 这种算法使任务不断地尝试寻找更优的流动途径,得到更优的任务分配矩阵. 算法遍历方式是随机的,即任意

选择两个相邻工位进行任务的尝试性流动,同时设置循环次数上限值作为随机遍历的终止条件.

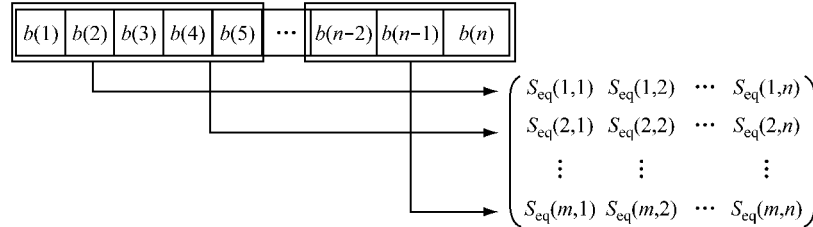


图 3 任务分配矩阵生成示意图

Fig.3 Formation of task allocation matrix

图 4 示例中①和②是可行的流动方案,序号 1 到 11 表示任务序号,矩形的长度表示任务操作时间. 任务流动调整的特点是在满足优先关系约束下,得到具有更高负载平滑系数的任务分配方案. 图 5

是图 4 中相邻任务在相邻工位间流动所对应的任务分配矩阵的变化. 任务分配矩阵与任务分配时间矩阵具有对应关系,当任务流动后根据式(4),(5),(12)可得到新的负载平滑系数.

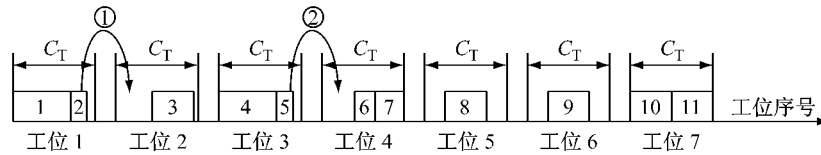


图 4 任务回溯示意图

Fig.4 Task backtracking

$$S_{cq} = \begin{pmatrix} 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 & 7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 11 & 0 \end{pmatrix}$$

图 5 任务分配矩阵中的任务流动

Fig.5 Task flow in task allocation matrix

3 案例分析

某装配流水线共有 7 个工位,29 个装配任务. 装配任务需要满足表 2 中所示的先后次序,同时任务 1,2 只能分配到工位 1 或 2 上,任务 8 只能分配到工位 3 或 4 上. 工位 5 只能接受任务 {2,3,4,6,8,10,11,12,13,14,15,16,17,18,20,21,22,24,25,26,29}, 工位 7 上只能接受任务 {1,2,3,4,5,6,7,8,9,12,13,21,22,23,23,24,25,26,27,28,11,29}.

在 Win7 64 位操作系统、8G 内存和 Intel(R) Core(TM)2 Duo CPU E8400(3.00GHZ)环境下,在 MATLAB 2013a 中编写相应算法程序. 算法中各参数设置如下: 种群规模 $S_{izepop} = 100$, 最大迭代次数 $M_{axgen} = 200$, 惯性权重 $\omega = 1$, 学习因子 $c_1 = 1.5, c_2 =$

1.5. 初步长 $\alpha = 1$, 控制精度因子 $\beta = 50$, 遍历循环次数上限 $N_c = 50$.

表 2 某流水线任务操作时间及紧前任务

Tab.2 Task operation time and immediate preceding task of an assembly line

任务序号	操作时间/s	紧前任务	任务序号	操作时间/s	紧前任务
1	7	—	16	7	8,14
2	19	—	17	14	11,13
3	15	1	18	17	16
4	5	3	19	10	15
5	12	4	20	16	17
6	10	2	21	1	19
7	8	—	22	9	18,21
8	16	5,6	23	25	20,22
9	2	7	24	14	23
10	6	9	25	14	1
11	21	8	26	2	2
12	10	7	27	10	26
13	9	5	28	7	23
14	4	10	29	20	24,26,27,28
15	14	10,12			

算法中没有采用精英保留策略,理论上组数越大,得到的最优节拍越好,算法计算的组数为 $\beta = 50$. 本文算法程序运算时间为 298 s. 图 6 表示组号与主

参考文献:

- [1] 鲁建厦,蒋玲玲,李修琳. 基于混合粒子群算法求解装配线第二类平衡问题[J]. 中国机械工程,2010(4):420.
LU Jiansha,JIANG Lingling,LI Xiulin. Hybrid particle swarm optimization algorithm for assembly line balancing Problem-2 [J]. Chinese Journal of Mechanical Engineering, 2010(4): 420.
- [2] 郑巧仙. 求解装配线平衡问题的蚁群算法研究[D]. 武汉:武汉大学,2013.
ZHANG Qiaoxian. Research on ant colony algorithm for assembly line balancing[D]. Wuhan: Wuhan University,2013.
- [3] 李明,唐秋华,郑巧仙,等. 第2类装配线平衡问题的改进规则组合算法[J]. 计算机集成制造系统,2015(1): 88.
LI Ming, TANG Qiuhua, ZHENG Qiaoxian, *et al.* Improved rules combination algorithm of type 2 assembly line balancing problem [J]. Computer Integrated Manufacturing Systems, 2015(1):88.
- [4] Mutlu Ö,Polat O,Supciller A A. An iterative genetic algorithm for the assembly line worker assignment and balancing problem of type-II [J]. Computers & Operations Research, 2013, 40: 418.
- [5] Petropoulos D, Nearchou A C. A particle swarm optimization algorithm for balancing assembly lines [J]. Dimitris I Petropoulos and Andreas C Nearchou, 2011,31(2):118.
- [6] 周亮. 装配线平衡的最优化模型与算法研究[D]. 南京:南京理工大学,2005.
ZHOU Liang. Study on assembly line balancing optimal models and algorithms [D]. Nanjing: Nanjing University of Science and Technology,2005.
- [7] Hamta N, FatemiGhomi S M T, Jolai F, *et al.* A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect [J]. International Journal of Production Economics,2013,141: 99.
- [8] Saif U,Guan Z,Liu W, *et al.* Pareto based artificial bee colony algorithm for multi objective single model assembly line balancing with uncertain task times[J]. Computers & Industrial Engineering,2014,76: 1.
- [9] 吴尔飞. 双边装配线平衡技术的研究[D]. 上海:上海交通大学,2009.
WU Erfei. Research on balancing two-sided assembly line[D]. Shanghai ;Shanghai Jiaotong University,2009.
- [10] 钱雄文. 多人共站装配线平衡问题的研究与优化[D]. 上海:上海交通大学,2012.
QIAN Xiongwen. Multi-manned assembly line balancing problem [D]. Shanghai ;Shanghai Jiaotong University,2012.
- [11] 徐立云,徐昌飞,邓伟,等. 基于 SA-PSO 算法的发动机缸体机加工线平衡研究[J]. 农业机械学报,2014(2):16.
XU Liyun, XU Changfei, DENG Wei, *et al.* Machining line balancing of engine cylinder block based on SA-PSO algorithm [J]. Transactions of Chinese Society for Agriculture Machinery,2014(2): 16.
- [12] 王慈光. 编组站动态配流模型与算法研究[J]. 铁道学报,2004,26(1):1.
WANG Ciguang. Research on the model and algorithm of dynamic wagon-flow allocating in a marshalling station [J]. Journal of the China Railway Society,2004,26(1):1.
- [9] Domarchi C, Tudela A, Gonzalez A. Effect of attitude, habit and affective appraisal on mode choice: An application to university workers [J]. Transportation Science, 2008, 35: 585.
- [10] Choo S, Mokhtarian P L. What type of vehicle do people drive? The role of attitude and lifestyle in influencing vehicle type choice[J]. Transportation Research; Part A, 2004, 38: 201.
- [11] Lam W H K, Li Z C, Huang H J, *et al.* Wong. Modeling time-dependent travel choice problems in road networks with multiple user classes and multiple parking facilities [J]. Transportation Research; Part B, 2006, 40: 368.
- [12] Li Z C, Lam W H K, Wong S C, *et al.* Reliability evaluation for stochastic and time-dependent networks with multiple parking facilities [J]. Networks and Spatial Economics, 2008, 8(4):355.
- [13] Bhat C R. Analysis of travel mode and departure time choice for urban shopping trips [J]. Transportation Research: Part B, 1998, 32: 361.
- [14] Wu J H, Florian M, Marcotte P. Transit equilibrium assignment: A model and solution algorithms [J]. Transportation Science, 1994, 28:193.
- [15] Lo H K, Yip C W, Wan K H. Modeling transfer and non-linear fare structure in multi-modal network [J]. Transportation Research; Part B, 2003, 37:149.
- [16] de Cea J, Fernandez E. Transit assignment for congested public transport systems: An equilibrium model [J]. Transportation Science, 1993, 27:133.
- [17] Lo H K, Yip C W, Wan K H. Modeling competitive multi-modal transit services: A nested logit approach [J]. Transportation Research; Part C, 2004, 12:251.

(上接第 241 页)