

# 双臂式双旋转头贴片机贴装效率优化研究

余松乐 肖曙红 林德育

(广东工业大学机电工程学院 广东 广州 510006)

**摘要** 为了优化双臂式双旋转头贴片机的贴装路径,提高贴片机的贴装效率,缩短 PCB 板的贴装时间,分析双臂式双旋转头贴片机的贴装路径优化问题类似于 TSP 问题。在考虑实际贴装过程的基础上,将贴装过程分为单贴片头内部的吸取贴装循环和两个贴片头之间的吸取贴装循环,并建立相对应的数学模型,提出基于取贴循环的蚁群算法对目标函数进行求解。理论分析和实验结果表明,改进的蚁群算法不仅符合贴片机的实际工作情况,提高了贴片机的贴装效率,而且还有效地克服传统算法中约束条件较多的缺点。

**关键词** 双臂式双旋转头贴片机 TSP 问题 蚁群算法 路径优化

中图分类号 TP391.9 文献标识码 A DOI:10.3969/j.issn.1000-386x.2019.09.026

## OPTIMIZATION OF MOUNTING EFFICIENCY OF DOUBLE-ARM DOUBLE-ROTATING HEAD MOUNTER

Yu Songle Xiao Shuhong Lin Deyu

(School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou 510006, Guangdong, China)

**Abstract** We aim to optimize the mounting path of the double-arm double-rotating head mounter, improve the mounting efficiency of the mounting machine and shorten the mounting time of PCB board. The optimization problem of the mounting path of the double-arm double-rotating head mounter is similar to the TSP problem. Considering the actual mounting process, the mounting process was divided into two parts: the internal absorption and mounting cycle of the single patch head; the absorption and mounting cycle between the two patch heads. We established the corresponding mathematical model and proposed the ant colony algorithm based on the extraction and mounting cycle to solve the objective function. Theoretical analysis and experimental results show that the improved ant colony algorithm conforms to the actual work situation of the mounting machine, improves the mounting efficiency and effectively overcomes the shortcomings of traditional algorithms with many constraints.

**Keywords** Double-arm double-rotating head mounter TSP problem Ant colony algorithm Path optimization

## 0 引言

随着对贴装效率的要求越来越高,单臂型贴片机已经不能满足要求,多臂式多头拱架型贴片机成为当前主流的贴片机。该类贴片机在针对一块 PCB 板贴装时只是两个贴装头交替贴装,实质上是一台双臂式或者多台双臂式串联而成,所以本文主要针对双臂式

双旋转头拱架型贴片机的贴装质量进行优化。贴片机的路径优化问题类似于 TSP 旅行商问题,即假设有一个旅行商人要拜访  $n$  个城市,每个城市只能拜访一次,最后要回到原来出发的城市,路径的选择目标是要求得的路径路程为所有路径之中的最小值。相对于模拟退火算法、遗传算法、蚁群算法解决此类问题效果更好<sup>[1]</sup>。单臂式拱架型贴片机在贴装过程中要考虑到贴片头的整体运动和自身的旋转运动、供料器的摆放位

置、元件的贴装顺序等问题,建立的数学模型具有高维数、离散性、非线性的特点<sup>[2]</sup>。

目前对贴片机的路径优化问题,很多学者都进行了研究。文献[3]提出了一种整体模型使用启发式优化算法对多臂式拱架型贴片机进行优化研究;文献[4]提出了一种分层规划方法去提高多头贴片机的贴装效率;Liu等<sup>[5]</sup>同样提出一种启发式优化算法;董娟娟等<sup>[6]</sup>则提出一种改进的遗传算法对多臂式多头拱架型贴片机进行优化研究;张坤等<sup>[7]</sup>采用蚁群算法对多头垂直贴片机进行贴装优化;臧明相等<sup>[8]</sup>提出一种基于分层混合蛙跳的多头贴片机的贴装调度算法。以上算法对多臂式多头贴片机的数学模型描述过于简单,数学模型只考虑贴片头的串行关系,即简单地将每个贴片头的贴装时间叠加起来为贴片机的整体贴装时间,没有反映出贴片头之间的并行关系,无法反映出真实的贴装情况。双臂式双旋转头拱架型贴片机不仅需要考虑一个贴片头内部取贴循环的优化,还需考虑两个贴片头之间的顺序优化。

本文在结合双臂式双旋转头拱架型贴片机的实际贴装情况下建立了合适的数学模型,然后结合实际贴装过程,提出了基于取贴循环优化的改进蚁群算法对贴装顺序进行优化求解,不仅提高了贴片机的贴装效率,而且还有效地克服了传统算法中约束条件较多的缺点。

## 1 双臂式双旋转头拱架型贴片机贴装过程

双臂式双旋转头拱架型贴片机的贴装过程为:给定一组贴片元件的顺序,旋转的贴片头移动到供料器上方,贴片头上的第1个吸嘴吸取一个元件,然后旋转一定角度到第2个吸嘴吸取元件。重复上述操作,直到第12个吸嘴吸取元件,然后贴片头运动到上视相机上方对元件进行角度和位置的识别矫正,最后移动到PCB板上进行贴装。此过程称为单贴片头的内部取贴循环,如图1所示。此类贴片机主要由供料器、带吸嘴的贴片头转盘(每个贴装头安装12个吸嘴)、PCB板组成。在工作过程中,当贴片头1处于吸取元器件状态时贴片头2处于贴装元器件状态,贴片头2处于吸取元器件状态时贴片头1处于贴装元器件状态,这样两个贴片头之间形成一个取贴循环的过程,如图2所示。

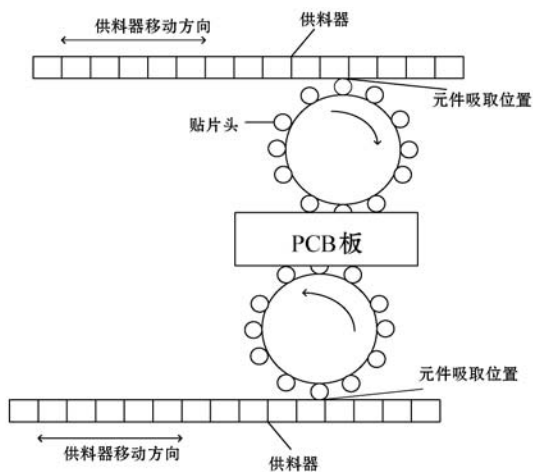


图1 双臂式双旋转头拱架型贴片机贴装示意图

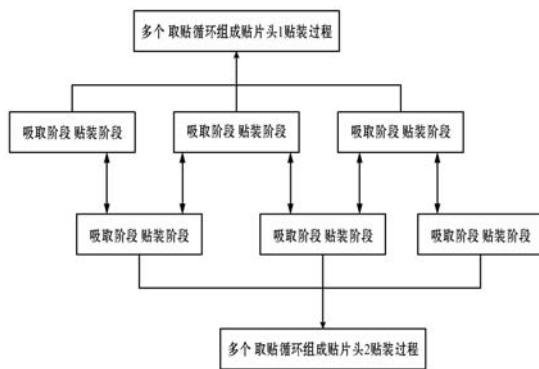


图2 取贴循环示意图

## 2 基于取贴循环的蚁群算法优化

文献[1]已表明蚁群算法解决TSP问题的效果较其他算法好,故本文在蚁群算法的基础上进行改进。双臂式双旋转头拱架型贴片机整体的路径优化较为复杂,因此可以将其简单化。根据实际的贴装循环过程将其分为两个部分进行路径优化。第一部分是单个贴片头内部的吸取和贴装元器件的循环路径优化,第二部分是两个贴片头之间的吸取和贴装元器件的循环路径优化。这两个过程相互之间有联系又有区别,且都影响着贴片机的贴装效率,因此应使这两个贴装循环的路径最优化。本文针对这两个过程建立了相对应的目标函数,并利用改进的蚁群算法对目标函数进行了求解。

### 2.1 取贴循环内部的路径优化

由于两个贴片头的贴装过程是相同的,所以这里只对一个贴片头进行研究。

1) 目标函数的确定。设待贴元件的种类为 $m$ ,数量为 $n$ ,送料器的个数为 $L$ ,转盘上贴装头个数为 $M$ ,吸取贴装循环的次数为 $C$ ,贴片头旋转一个贴装头的时

间为  $t_1$ , 工业相机检测一个元器件的时间为  $t_2$ , 供料器第  $i$  个元器件的准备时间为  $t_i^3$ , 吸嘴完成拾取第  $i$  个元器件的时间为  $t_i^4$ , 吸嘴完成第  $i$  个元件贴装的时间为  $t_i^5$ , 贴片头吸取和贴装第  $i$  个元器件的速度为  $v_i$ , 则贴片头从吸取元件  $i$  开始到吸取完元件  $j$  结束之间的距离  $S_{ij}$  可以表示为:

$$S_{ij} = \sum_{L_1=L_2}^m Q_{(t(i),L_1)} Q_{(t(j),L_2)} d(L_1, L_2) \quad (1)$$

式中:  $Q_{(t,L)} = \begin{cases} 1 & \text{元件类型 } t \text{ 位于供料器 } L \text{ 中} \\ 0 & \text{其他} \end{cases}$ ,  $t(i)$  表示元件  $i$  所属的类型,  $t(i) \in \{1, 2, \dots, m\}$ ,  $d(L_1, L_2)$  表示两个供料器之间的距离。则贴片头吸取元件  $i, j$  之间的时间  $T_{(i,j)}^1$  可以表示为:

$$T_{(i,j)}^1 = \begin{cases} \max(t_1, t_j^3) + t_j^4 & i, j \text{ 处于相同供料器} \\ \max(t_1, W_{ij} S_{ij} / v_j) \times X_{ij} + t_j^4 & i, j \text{ 处于不同供料器} \end{cases} \quad (2)$$

式中:  $W_{ij} = \begin{cases} 1 & i, j \text{ 位于不同供料器} \\ 0 & i, j \text{ 位于相同供料器} \end{cases}$   
 $X_{ij} = \begin{cases} 1 & \text{吸取 } i \text{ 之后紧接着吸取 } j \\ 0 & \text{其他} \end{cases}$

贴片头贴装元件  $i$  开始到贴装完元件  $j$  结束之间的时间  $T_{(i,j)}^2$  为:

$$T_{(i,j)}^2 = \max(t_1, D_{ij} / v_j) \times X_{ij} + t_j^5 \quad (3)$$

式中:  $D_{ij}$  表示元件  $i, j$  之间的贴装距离。以上三部分组成一个取贴循环。则一个取贴循环时间  $T^3 = T_{(i,j)}^1 + T_{(i,j)}^2 + t_1 + t_2$ , 第  $k_1$  ( $k = (1, 2, \dots, C)$ ) 个循环的时间为:

$$T_{k1} = \sum_{i=1}^{k1} T_{(i,i+1)}^1 + \sum_{i=1}^{k1} T_{(i,i+1)}^2 + (t_1 + t_2) \times k_1 \quad (4)$$

所以单贴片头内部取贴循环的目标函数为:

$$T_1 = \min\left(\sum_{k_1=1}^C T_{k1}\right) \quad (5)$$

2) 种群规模的确定。贴装元件的数量  $n$  确定了蚁群算法的种群规模  $m$ , 在保证最低性能的情况下应当适量减少蚁群的数量, 本文选择的种群规模为  $m = 0.6n$ 。

3) 蚂蚁的选择转移策略。在解决 TSP 问题时, 蚁群算法中的每只蚂蚁选择下一个城市是根据状态转移概率选择的, 如下式所示:

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in allowed_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)} & j \in allowed_k \\ 0 & \text{其他} \end{cases} \quad (6)$$

式中:  $allowed_k$  表示蚂蚁  $k$  下一步允许移动的城市;  $\eta$  为启发性信息;  $\tau$  为由第  $i$  个城市到第  $j$  个城市的路径的信息素强度;  $\alpha$  为信息启发式因子,  $\beta$  为期望启发因子。但贴片头在贴装时选择下一个元件时要考虑吸取元件和贴装元件的时间, 两者的侧重比例不同, 因此引入一个参数  $\delta$  表示两者之间受重视的程度, 所以状态转移概率为:

$$P_{ij}^k(t) = \begin{cases} \frac{\delta \tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in allowed_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)} + \frac{(1-\delta) \tau_{ij}^\alpha(t) \mu_{ij}^\beta(t)}{\sum_{s \in allowed_k} \tau_{is}^\alpha(t) \mu_{is}^\beta(t)} & j \in allowed_k \\ 0 & \text{其他} \end{cases} \quad (7)$$

式中: 启发函数  $\eta_{ij} = 1/T_{ij}^1$ ,  $T_{ij}^1$  计算如式(2), 启发函数  $\mu_{ij} = 1/T_{ij}^2$ ,  $T_{ij}^2$  计算如式(3)。参数  $\delta = (0, 1)$ , 当贴片元件数量多、种类少时, 贴装时间多, 取  $\delta < 0.5$ ; 当元件数量少、种类多时, 吸取时间多, 取  $\delta \geq 0.5$ 。

4) 新的信息素更新策略。局部更新策略: 每个贴片头有 12 个吸嘴, 一个循环内可实现 12 个元件的取贴, 根据这个特点, 可得蚁周系统的模型和  $t$  时刻在路径  $(i, j)$  路径上的更新策略分别为:

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & \text{若 } Ant_k \text{ 在本次循环中经过路径 } (i, j) \\ 0 & \text{其他} \end{cases} \quad (8)$$

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{M_1}{n} \times \frac{Q_1}{T_k^0} & \text{若第 } k \text{ 只蚂蚁在本次取贴循环中经过路径 } (i, j) \\ 0 & \text{其他} \end{cases} \quad (9)$$

式中:  $\Delta \tau_{ij}^k(t)$  为本次循环中的信息素增量,  $L_k$  为蚂蚁  $k$  在本次循环中经过的路径总长,  $Q$  为局部信息素强度,  $n$  为元件数量,  $T_k^0$  为第  $k$  个取贴循环的时间, 计算如式(4)所示。

全局更新策略: 为了使算法在尽可能大的范围内尽可能快地收敛到全局最优解, 引入一个惩罚因子  $\omega_1$ , 对全部解区间的信息素进行自适应的更新调整。  $t$  时刻在路径  $(i, j)$  上的更新规则为:

$$\Delta \tau_{ij}^k(t) = \frac{Q \times \omega_1}{T_{best}^k}$$

式中:  $Q$  为全局信息素强度,  $Q$  的值要根据贴装元件规模的大小来调整;  $\omega_1 = T_{best}^k / T^k$ ,  $T_{best}^k$  为当前最优解,  $T_{best}^k$  值越小结果越优; 表示本次迭代中得到的解, 当本次迭代中的值优于当前最优解时,  $\omega_1 > 1$  并更新当前最优解, 反之  $\omega_1 \leq 1$ 。

## 2.2 取贴循环间的路径优化

取贴循环间连接的主要因素有:两个贴片头从前一个取贴循环的最后一个元件的贴放位置到下一个循环第一个元件的吸取位置的移动时间  $t_3$ ,两个贴片头同时吸取耗时的时差  $t_4$ 。 $t_3^1 = s_1 \div v_1, t_3^2 = s_2 \div v_2$ ,其中: $s_1, s_2$ 分别表示贴片头 1 和贴片头 2 从前一个取贴循环的最后一个元件的贴放位置到下一个取贴循环第一个元件的吸取位置之间的距离, $v_1, v_2$ 分别表示贴片头 1、贴片头 2 的移动速度。前面已经分析了单个贴片头从第  $i$  个吸取完成到第  $j$  个吸取完成时间的目标函数  $T_{ij}^1$ 、从第  $i$  个贴装完成到第  $j$  个贴装完成时间的目标函数  $T_{ij}^2$ ,而两个贴片头的贴装过程完全相同,所以目标函数也相同,设为  $T_{ij}^3, T_{ij}^4$ ,所以交叉循环的目标函数可以写成:

$$T_k = \max\left(\sum_{i=1}^{k_1} T_{(i,i+1)}^1, \sum_{j=1}^{k_1} T_{(j,j+1)}^2 + (t_1 + t_2) \times k_1\right) + \max(t_3^1, t_3^2) + \max\left(\sum_{i=1}^{k_2} T_{(i,i+1)}^4, \sum_{i=1}^{k_2} T_{(i,i+1)}^3 + (t_1 + t_2) \times k_2\right) + t_4 \quad (10)$$

式中: $k_1, k_2$ 分别表示贴片头 1、贴片头 2 的循环次数, $k$ 为交叉循环的次数。

所以贴片机的整个贴装的目标函数可以写成:

$$T_0 = \min\left(\sum_{k=1}^C T_k\right) \quad (11)$$

目标函数满足的约束条件为  $\sum_{k=1}^c (k_1 + k_2) = n$ 、

$\sum_{i=j=1}^{n_1} Y_{ij}^1 - 1 = \sum_{i=j=1}^{n_2} Z_{ij}^2 = C - 1, \sum_{i=j=1}^{n_2} Y_{ij}^2 = C$ ,其中: $i$ 为本次交叉循环贴片头 1 吸取的最后一个元件, $j$ 为下次交叉循环贴片头 1 吸取的第一个元件时, $Y_{ij}^1 = 1$ ;其他  $Y_{ij}^1 = 0$ 。 $Y_{ij}^2$ 则表示贴片头 2, $i$ 为本次交叉循环贴片头 2 贴装的最后一个元件, $j$ 为下次交叉循环贴片头 2 贴装的第一个元件时, $Z_{ij}^2 = 1$ ;其他  $Z_{ij}^2 = 0$ 。

种群规模的选择和前面相同,蚂蚁的选择转移策略选择一般的状态转移概率选择,如式(6)所示。全局信息素的更新采用基于蚁周系统模型进行信息素的更新,更新公式为式(8),所以信息素的增量为:

$$\Delta\tau_{ij}^k(t) = \frac{Q_s}{T_0} \quad (12)$$

## 2.3 改进后的算法特点

改进后的算法的特点主要有:(1)在进行取贴循环内部元器件顺序优化时,依据元器件的分组,蚂蚁在

选择下一个元器件时会先选择同组元器件,减少了蚂蚁搜索元器件的数量,缩短了算法的搜索时间,提高了算法的效率。(2)采用基于取贴循环的蚁群算法将贴片机的整体优化复杂问题分解为多个简单的子优化问题,缩小了求解优化问题的规模,提高了最优路径的搜索效率。(3)局部和全局相结合的信息素更新方式,避免了只有最优蚂蚁在每次循环后更新信息素致使陷入局部最优解。

## 3 实验结果分析

本文实验将对表 1 中 5 块 PCB 板的元器件进行路径优化。实验过程中,设基本蚁群算法的参数  $Q, \alpha, \beta$ 、迭代次数与改进算法相同,蚂蚁种群规模相同为  $m = 0.6 \times n$ ,其中  $n$  为元器件个数,基本蚁群算法状态转移概率公式则采用式(6),信息素更新采用全局更新规则,贴片机的基本性能参数与改进算法相同。

表 1 PCB 板元件特性

PCB	元器件总数	贴片头	元件数	分类数	分组数
1	88	贴片头 1	44	5	3
		贴片头 2	44	5	3
2	158	贴片头 1	82	9	4
		贴片头 2	76	8	4
3	207	贴片头 1	103	13	4
		贴片头 2	104	12	3
4	485	贴片头 1	240	24	6
		贴片头 2	245	25	5
5	720	贴片头 1	360	36	10
		贴片头 2	360	36	10

双臂式双旋转头贴片机的一些参数  $t_1 = t_2 = 10 \text{ ms}, v_1 = v_2 = 1 \text{ 000 mm/s}$ ,为了体现元器件尺寸对贴装性能的影响, $t_1, t_2$ 以 20% 的比例增加, $v_1, v_2$ 以 20% 的比例减小。实验中采用局部和全局信息素相结合的更新方式。程序的开发环境为 Visual Studio2010/MFC,程序的测试环境为:双核 2.8 GHz CPU,3 GB 内存,Windows 7 的操作系统。

以表 1 中 2 号和 5 号 PCB 板为例,使用改进的蚁群算法路径优化效果如图 3 所示,图中上下两排点表示供料器的位置,灰色的点表示贴片头 1 贴装的元件,黑色的点表示贴片头 2 贴装的元件,不同大小的点表示不同尺寸的元件。(a)和(b)表示 2 个贴片头单独贴装的路径图,(c)和(d)为 2 个贴片头一起贴装的路径图。

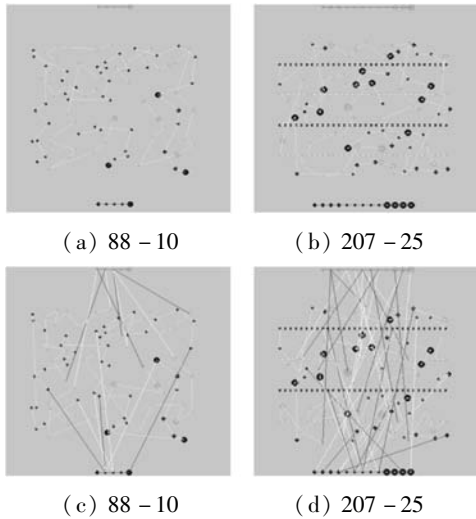


图3 路径优化效果图

为了验证本文算法的优越性,将本文改进的蚁群算法与基本的蚁群算法、遗传算法分别对表1中5块PCB板进行20次计算,计算的最优、最差、平均解如表2所示。

表2 算法结果比较表

PCB	算法	最优/s	最差/s	平均/s
1	蚁群算法	5.591	5.628	5.612
	遗传算法	8.454	9.254	8.745
	本文算法	5.468	5.548	5.495
2	蚁群算法	8.872	8.932	8.905
	遗传算法	11.546	15.748	13.564
	本文算法	8.792	8.822	8.836
3	蚁群算法	12.992	13.334	13.152
	遗传算法	18.638	24.810	20.256
	本文算法	11.659	12.057	11.873
4	蚁群算法	12.996	13.334	13.152
	遗传算法	73.115	78.357	75.225
	本文算法	27.235	27.696	27.465
5	蚁群算法	85.076	88.958	86.245
	遗传算法	158.104	170.778	165.268
	本文算法	44.953	45.845	45.405

## 4 结 语

本文分析了双臂式双旋转头拱架型贴片机的实际贴装过程,将其复杂的路径优化分为简单的单贴片头内的取贴循环的路径优化和双头之间的取贴交叉循环的路径的优化,建立了对应的模型。利用基于取贴循环的蚁群算法对模型分别进行求解。实验结果表明,本文提出的基于取贴循环的蚁群算法符合双臂式双旋转头贴片机的实际贴装过程,求得的最优解明显优于

其他算法的最优解,缩短了PCB板的贴装时间,大幅提高了贴装效率。

## 参 考 文 献

- [1] 潘云峰. 垂直旋转式高速贴片机贴装路径优化研究[D]. 镇江:江苏大学,2017.
- [2] 林德育,肖曙红,麦智伟. 水平旋转式贴片机贴装质量优化控制仿真[J]. 计算机仿真,2017,34(5):300-304.
- [3] Kumar R, Li H. Integer programming approach to printed circuit board assembly time optimization[J]. IEEE Transactions on Components, Packaging, and Manufacturing Technology: Part B, 1995, 18(4):720-727.
- [4] Lee S H, Lee B H, Park T H. A hierarchical method to improve the productivity of a multi-head surface mounting machine[C]//Proceedings of the 1999 IEEE International Conference on Robotics Automation. IEEE, 1999.
- [5] Liu H M, Hu Y M, Yuan P, et al. A Heuristic Optimization Algorithm for Multi-head Moulder[C]//2007 IEEE 22nd International Symposium on Intelligent Control. IEEE, 2007: 279-384.
- [6] 董娟娟. 多头拱架型高精度贴片机的贴装工艺优化[D]. 合肥:合肥工业大学,2013.
- [7] 张坤. 多头垂直旋转式贴片机的贴装工艺优化研究[D]. 西安:西安电子科技大学,2009.
- [8] 臧明相,杜鑫,李斌. 基于分层混合蛙跳的多头贴片机贴装调度算法[J]. 计算机集成制造系统,2013(11):2847-2853.
- [9] 陈铁梅,罗家祥. 改进禁忌搜索算法的贴片机的贴装过程优化[J]. 计算机工程与应用,2012,48(17):31-35.
- [10] 杜轩,李登桥,张屹. 基于混合遗传算法的多头动臂式贴片机的贴装过程优化研究[J]. 机床与液压,2014(1):40-47.

## (上接第105页)

- [15] 何所畏. 基于贝叶斯网络的冷水机组故障检测和诊断研究[D]. 西安:西安建筑科技大学,2017.
- [16] 吴欣. 基于改进贝叶斯网络方法的电力系统故障诊断研究[D]. 杭州:浙江大学,2005.
- [17] 胡玲玲. 基于贝叶斯网络的空气制动系统故障诊断的研究[D]. 北京:北京交通大学,2010.
- [18] 朱明敏. 贝叶斯网络结构学习与推理研究[D]. 西安:西安电子科技大学,2013.
- [19] 罗苏,吴锡坤. 铝型材加工实用技术手册[M]. 长沙:中南大学出版社,2006.
- [20] 许福玲,陈尧明. 液压与气压传动[M]. 武汉:华中科技大学出版社,2001.
- [21] 王双成. 贝叶斯网络学习、推理与应用[M]. 上海:立信会计出版社,2010.
- [22] 盛骤,谢式千,潘承毅. 概率论与数理统计[M]. 4版. 北京:高等教育出版社,2008.