

改进的基于蜜蜂进化型遗传算法和蚁群系统混合的元件贴装优化

王超学¹, 孙有田¹, 董 惠¹, 崔杜武²

(1 西安建筑科技大学 信息与控制工程学院, 陕西 西安 710055;

2 西安理工大学 计算机科学与工程学院, 陕西 西安 710048)

摘 要: 针对 PCB 板的表面贴装技术(Surface Mount Technology, SMT)优化问题, 提出一种基于蜜蜂进化型遗传算法和蚁群系统的混合智能算法(the Hybrid Intelligent Algorithm based on Bee Evolutionary Genetic Algorithm and Ant Colony System, BAHA). 该算法的关键有 4 点: ①通过两个种群的融合实现信息共享, 提高算法的收敛速度; ②采用改进的 OX 的交叉算子, 合理保留优秀个体基因的排列顺序; ③加入局部搜索算子, 在当代最优解附近进行更加精细的搜索; ④信息素重置防止陷入局部最优解. 用 TSP30 问题、eil51 问题与相关文献进行对比测试, 仿真结果表明 BAHA 收敛速度快, 寻优能力强. 通过对 5 种不同 PCB 板的元件贴装顺序进行优化计算, 结果表明, BAHA 能有效的提高贴装效率.

关键词: 表面贴装技术; 蜜蜂进化型遗传算法; 蚁群系统; OX 交叉; 局部搜索; 信息素重置

中图分类号: TP202

文献标识码: A

文章编号: 1000-7180(2012)08-0158-06

The Component Mounting Optimization Based on Improved Bee Evolutionary Genetic Algorithm and Ant Colony System

WANG Chao-xue¹, SUN You-tian¹, DONG Hui¹, CUI Du-wu²

(1 School of Information and Control Engineering, Xi'an University of Architecture & Technology, Xi'an 710055, China;

2 School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China)

Abstract: To optimize the Surface Mount Technology (SMT) of PCB card, a hybrid intelligent algorithm based on bee evolutionary genetic algorithm and ant colony system (BAHA) is proposed. The key of the hybrid intelligent algorithm lies in improving the convergence speed by the combination between the two population; the improved OX crossover operator retained the sequence of the good genes; introducing a Local search operator, which has more elaborate search ability in the neighborhood of the iteration-best; pheromone resetting is used to jump from local optimal solution. The test results on TSP 30 problem, eil51 problem and the component mounting sequence problem of 5 PCB cards show that BAHA has good global search ability and fast convergence rate, and promote the mounting efficiency effectively.

Key words: surface mount technology; bee evolutionary genetic algorithm; ant colony system; OX crossover; local search; pheromone resetting

1 引言

随着现代电子技术的发展, 表面贴装技术(Surface Mount Technology, SMT)已成为电子产品生

产中的主流技术, 贴片机作为表面贴装生产线中的重要设备, 其贴装速度对整条生产线的生产效率有着重要的影响. SMT 的优化一般分为两个子问题来解决: 一是喂料器分配固定的元器件贴装优化, 通常

收稿日期: 2012-01-06; 修回日期: 2012-03-16

基金项目: 国家自然科学基金(31170393); 陕西省自然科学基金(2012JM8023); 陕西省教育厅自然科学基金专项(12JK0726)

作为带约束的 TSP (travelling salesman problem) 问题来解决^[1-2];二是元器件贴装顺序确定的喂料器分配优化,通常作为 QAP (quadratic assignment problem) 问题来解决^[3]. 对于前者,文献[4]运用遗传算法求解,遗传算法具有全局寻优能力强、鲁棒性强等优点,但其对反馈信息利用不够,当求解到一定范围时做大量冗余迭代,求解效率不理想;文献[5]用蚁群算法求解,蚁群算法通过信息素的积累反馈信息收敛于最优解,具有较好的寻优能力,但由于初期信息素匮乏,算法初期求解速度较慢.

本文在喂料器分配固定的情况下,针对元器件贴放顺序优化模型,利用蜜蜂进化型遗传算法^[6] (Bee Evolutionary Genetic Algorithm, BEGA) 的全局收敛性,结合蚁群系统^[7] (Ant Colony System, ACS) 的信息反馈、学习机制,提出了一种基于蜜蜂进化型遗传算法和蚁群系统的混合智能算法 (the Hybrid Intelligent Algorithm based on Bee Evolutionary Genetic Algorithm and Ant Colony System, BAHA), 仿真结果表明,本文提出的算法收敛速度较快,求解质量较高,能有效的优化元器件贴装顺序,提高贴片机生产效率.

2 模型描述

完成一块 PCB 板的贴装要经过多个贴装循环,每次循环中,贴装头上的吸嘴在喂料器上拾取元器件,经由相机对元器件识别校正后,移到 PCB 板上方将元器件贴到正确的位置,因为吸嘴在拾取或者贴装过程中有停顿减速时间,贴装头移动速度也因元器件的体积重量等有所不同,且贴装时间主要取决于贴装头走过的路程长度,所以本文以贴装头走过的路程作为优化的目标^[1].

设整个贴装过程有 N 个贴装循环,即贴装头需要取料 N 次,每一个贴装循环的过程都是贴装头选取合适的元件,再依次将贴装头上的元件贴到对应贴装点上. 在第 i 个贴装循环中,贴装头走过的路程 l_i 如式(1)所示:

$$l_i = l_i^a + l_i^b + l_i^c + l_i^d \quad (1)$$

式中, l_i^a 为贴装头从上一轮贴装循环的结束位置到本轮贴装循环的第一个喂料器位置的路程长度, l_i^b 是贴装头从本轮贴装循环的第一个喂料器位置到最后一个喂料器位置的路程长度, l_i^c 是贴装头从本轮贴装循环的最后一个喂料器位置到本轮循环的第一个贴装位置的路程长度, l_i^d 是从本轮循环的第一个贴装位置到最后一个贴装位置 (即下一轮的开始位

置) 的路程长度.

整个贴装过程的路程长度如式(2)所示:

$$L = \sum_{i=1}^N l_i = \sum_{i=1}^N (l_i^a + l_i^b + l_i^c + l_i^d) \quad (2)$$

$A = \{a_1, a_2, \dots, a_i, \dots, a_N\}$, $B = \{b_1, b_2, \dots, b_i, \dots, b_N\}$, $C = \{c_1, c_2, \dots, c_i, \dots, c_N\}$, 其中, a_i 为贴装循环 i 中需用元件集合, b_i 贴装循环 i 中拾取循序, c_i 为贴装循环 i 中的贴装顺序, 本文的贴装优化就是找到最好的 A, B, C , 得到最好的 L ; 该模型的原型可以看做为带约束的 TSP 问题.

3 算法部分

本文在前人基础上,提出了一种基于蜜蜂进化型遗传算法和蚁群系统的混合智能算法 (BAHA). 该算法在寻优过程中分别用改进的 BEGA 和改进的 ACS 从两条路径进行搜索构成新的总种群,实现种群间信息共享; BEGA 以总种群中的最优个体作为蜂王与雄蜂进行交叉操作,进化过程中加入的随机种群保持了种群多样性,提高算法的勘探能力,防止早熟. 运用总种群中最优个体更新全局信息素后进行蚂蚁搜索,缩短了 ACS 初期搜索时间;同时利用 ACS 的信息反馈和学习机制,提高整个算法的收敛速度;信息素重置防止陷入局部最优解. 引入局部搜索算子,在每代最优解附近进行更加精细的搜索,进一步增强算法的开采能力. 算法具体实现如下所示.

3.1 改进的蚁群系统

ACS^[7] 是意大利学者 M. Dorigo 于 1997 年提出的,该算法是性能最好的蚁群优化算法之一,本文在 ACS 的基础上,引入信息素重置操作,防止算法陷入局部最优解.

3.1.1 蚂蚁搜索

蚁群规模为 m . 蚂蚁 k 下一个访问的城市要从蚂蚁 k 未访问过的城市选取,蚂蚁 k 需根据当前城市到备选城市之间的路径长度和该路径的信息素值选取下一个城市. 设蚂蚁 k 当前所在城市为 i ,按照状态转换规则选择下一个城市 j ,如式(3)和(4)所示:

$$\begin{aligned} & \text{在 } t \text{ 时刻,当 } q < q_0 \text{ 时,} \\ & s = \arg \max_{j \in \text{allowed}_k} \tau_{ij}^a(t) \cdot \eta_{ij}^b(t) \end{aligned} \quad (3)$$

否则,

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^a(t) \cdot \eta_{ij}^b(t)}{\sum_{j \in \text{allowed}_k} \tau_{ij}^a(t) \cdot \eta_{ij}^b(t)} & \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

式中, α 为信息素启发式因子, β 为期望启发式因子, τ_{ij} 为城市 i 和 j 之间的信息素, η_{ij} 为启发因子, 是城市 i 和 j 之间距离的倒数, allowed_k 为候选城市集合, q 是在 $[0, 1]$ 区间均匀分布的随机数, q_0 为设定的阈值, 本文将 q_0 设为 0.75.

3.1.2 更新信息素

每次迭代开始, 将总种群中最优个体选出并用该个体进行全局信息素更新, 如式(5)和(6)所示:

$$\tau_{ij}(t+1) = (1 - \rho_1) \cdot \tau_{ij}(t) + \rho_1 \cdot \Delta\tau'_{ij} \quad (5)$$

$$\Delta\tau'_{ij} = \begin{cases} (\text{Lbest})^{-1}, & \text{if } (ij) \in \text{best} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

式中, τ_{ij} 为路径 (i, j) 的信息素, ρ_1 为全局挥发因子, Lbest 为最优个体的适应度, best 为最优路径.

蚂蚁 k 每经过一条路径, 则按照式(7)对该路径进行局部信息素更新:

$$\tau_{ij}(t+1) = (1 - \rho_2) \cdot \tau_{ij}(t) + \rho_2 \cdot Q_0 \quad (7)$$

式中, Q_0 为初始信息素, ρ_2 为局部挥发因子.

3.1.3 重置信息素

ACS 利用正反馈原理强化已找到的次优解, 当迭代到一定代数后, 次优解路径上的信息素不断得到强化, 使得蚂蚁集中在几条路径上, 从而陷入局部最优解, 为了防止早熟现象的产生, 本文算法加入重置信息素操作, 即当总种群的最优个体在连续 20 次迭代中没有变化时, 将全部信息素重设为 Q_0 .

3.2 改进的蜜蜂进化型遗传算法

BEGA^[6] 将种群中的最优个体作为蜂王与被选个体(雄蜂)进行交叉操作, 增强了对种群最优个体包含信息的开采能力, 且在进化过程中引入了一个随机种群, 提高了算法的勘探能力. 本文在此基础上, 引入自适应交叉变异操作, 有利于算法开采与勘探的平衡, 同时改进更新种群操作, 保留优秀父代个体.

3.2.1 选择

采用轮盘赌选择从总种群中选取 $N * u/2$ 个个体, 再随机生成 $N * (1-u)/2$ 个个体, 作为父代个体组成种群 POPb, 该种群规模为 $N/2$, 其中 N 表示 BEGA 的种群规模, $N = NN - m$, NN 表示总种群规模, $0 \leq u \leq 1$; u 若过小, 容易引起种群退化, 若过大, 种群多样性降低, 易陷入局部最优解, 本文将 u 设置为 0.5.

3.2.2 交叉

将总种群中的最优个体作为蜂王与父代个体(雄蜂)进行交叉操作, 充分利用总种群的共享信息. 交叉操作采用改进的 OX 顺序交叉算子, 交叉方法

以下面两个个体为例说明($/ * /$ 表示交叉的部分): 父个体 $p1: (1\ 2\ 3/4\ 5\ 6/7\ 8\ 9)$; 父个体 $p2: (6\ 3\ 5/1\ 9\ 7/2\ 4\ 8)$. 将父体 2 的交叉部分复制到父体 1 交叉部分前, 得到 $l1: (1\ 2\ 3/1\ 9\ 7/4\ 5\ 6\ 7\ 8\ 9)$; 将父体 1 的交叉部分复制到父体 2 交叉部分前, 得到 $l2: (6\ 3\ 5/4\ 5\ 6/1\ 9\ 7\ 2\ 4\ 8)$. 将中间插入的部分固定, 对其余部分进行扫描, 去掉与插入部分相同的城市码, 如 $l1$ 中间的 $(1\ 9\ 7)$ 固定, 其余部分进行扫描, $l1$ 中第 1、10、12 个编码为城市码 1、7、9, 与插入部分城市码相同, 去掉. 扫描删除完成后得到子个体为: 子个体 $c1: (2\ 3\ 1\ 9\ 7\ 4\ 5\ 6\ 8)$; 子个体 $c2: (3\ 4\ 5\ 6\ 1\ 9\ 7\ 2\ 8)$.

交叉操作采用了自适应交叉算子, 交叉概率如式(8)所示:

$$P_c^t = \begin{cases} \frac{p_{c\max}}{1 + t/t_{\max}}, & p_c^t > p_{c\min} \\ p_{c\min}, & p_c^t \leq p_{c\min} \end{cases} \quad (8)$$

式中, P_c^t 为当前的交叉概率, $P_{c\max}$ 为最大交叉概率, $P_{c\min}$ 为最小交叉概率, t 为当前进化代数, $t_{\max}$ 为最大进化代数.

3.2.3 变异

算法中采用了自适应变异算子, 变异概率如式(9):

$$p_m^t = \begin{cases} \frac{p_{m\max}}{1 + t/t_{\max}}, & p_m^t > p_{m\min} \\ p_{m\min}, & p_m^t \leq p_{m\min} \end{cases} \quad (9)$$

式中, P_m^t 为当前的交叉概率, $P_{m\max}$ 为最大交叉概率, $P_{m\min}$ 为最小交叉概率, t 为当前进化代数, $t_{\max}$ 为最大进化代数.

3.2.4 更新种群

为保留父代个体中的优秀基因, 本文对种群更新进行了改进, 在交叉变异操作后, 得到的种群 POPc 规模为 N , 将种群 POPb 和种群 POPc 合并生成种群 POPd, 其种群规模为 $3 * N/2$, 并从种群 POPd 中采用轮盘赌选取 N 个个体组成种群 POPE, 用蜂王替换 POPE 中最差个体, 再将 POPE 与种群 POPa 合并生成总种群 POP, 其规模为 $N + m$ 个, 即 NN 个.

3.3 局部搜索算子

每次迭代得到当代最优个体以后, 采用局部搜索算子在当代最优解附近进行更精细的搜索, 加快算法的收敛速度. 局部搜索算子采用改进的 3-OPT 算法^[8]: 假设 $d(X, Y)$ 表示城市 X 和 Y 之间的距离, R_i 表示最优个体中第 i 个城市, 城市数目为

n , 则从 $[1, n]$ 之间随机选择 3 个点 i, j, h , 其中 $i < j < h$, 若 $d(R_i, R_{i+1}) + d(R_j, R_{j+1}) > d(R_i, R_j) + d(R_{i+1}, R_{j+1})$, 则将子集合 $\{R_{i+1}, \dots, R_j\}$ 中元素反向排列, 同理判断 $\{R_{j+1}, \dots, R_h\}$ 中元素是否需要反向, 将该操作重复五次; 然后对路径进行微调, 选择五个点 $w, w+1, w+2, w+3, w+4$, 对中间三个点进行全排列, 每排列一次, 计算从点 w 到点 $w+4$ 的路径距离, 从中选取路径最小的一个排列, 并将对应的点排列代替原来路径中的点排列。

3.4 算法流程

BAHA 流程如下所示:

Step 1 参数初始化, 信息素初始化, 随机生成初始总种群 POP, 其规模为 NN , 令迭代次数 $t=0$;

Step 2 计算总种群适应度, 最优个体作为蜂王;

Step 3 如满足停止准则, 输出结果并停止运行; 否则, 执行 Step 4;

Step 4 $t=t+1$, 用蜂王进行全局信息素更新;

Step 5 进行蚂蚁搜索, 蚂蚁每走一步, 进行信息素的局部更新, 所有蚂蚁遍历完后, 蚂蚁经过的 m 条路径作为 m 个个体组成种群 POPa;

Step 6 采用轮盘赌选择从总种群中选取 $N * u / 2$ 个个体, 再随机生成 $N * (1-u) / 2$ 个个体, 作为父代个体组成父代种群 POPb, 该种群规模为 $N/2$;

Step 7 用蜂王与种群 POPb 中的 $N/2$ 个个体进行交叉操作; 变异后生成子代种群 POPc, 种群规模为 N ;

Step 8 将种群 POPb 和种群 POPc 合并生成种群 POPd, 其种群规模为 $3 * N/2$, 并从种群 POPd 中采用轮盘赌选取 N 个个体组成种群 POPE;

Step 9 将种群 POPa 和种群 POPE 合并更新总种群 POP, 种群规模为 NN ;

Step 10 进行局部搜索以及信息素重置操作, 返回 Step 2.

4 仿真实验

在 MATLABR2009a 环境中, 蚁群系统 (Ant Colony System, ACS)、蜜蜂进化型遗传算法 (Bee Evolutionary Genetic Algorithm, BEGA) 和 BAHA 利用 TSP30 问题以及 eil51 问题对算法的有效性进行仿真验证, 分别独立测试 50 次, 并与文献[8-10]给出的数据进行对比. 参数设定为: 总种群规模 NN 为 110, 蜜蜂进化型遗传算法种群规模 N 为 100, 交叉概率初值为 0.9, 变异概率初值为 0.3, 蚂蚁数目 $m=10$, $\alpha=1$, $\beta=5$, $\rho_1=0.1$, $\rho_2=0.1$. 表 1、表 2 关

键字说明, Lbest 为最优长度, NC 为运算次数, Nc 为出现当前最优解的次数, ns 为出现最优解的迭代次数, La 为平均路径, Lb 为当前已知的最优解, TSP30 问题实验结果如表 1 所示.

表 1 TSP30 问题实验结果

算法	Lbest	NC	Nc	ns	La	Lb
文献[8]	423.7	/	/	52	/	
文献[9]	424.6	10	0	/	/	
ACS	423.7	50	2	109	429.8	423.7
BEGA	426.9	50	0	124	447.9	
BAHA	423.7	50	37	26	424.0	

由表 1 可知, 在 50 次独立实验中, BAHA 有 37 次能收敛到最优解 423.7, 且最快可在第 26 代收敛, 算法最优解的平均值为 424.0; 与 ACS、BEGA 相比, BAHA 具有较好的收敛速度和稳定性; BAHA 的收敛速度优于文献[8], 其搜索性能优于文献[9].

在 eil51 问题中常用整数方式求解, 即将城市间的距离长度四舍五入化为整数再求解, 其最优解为 426, 路径如图 1 所示, 实验结果如表 2 所示.

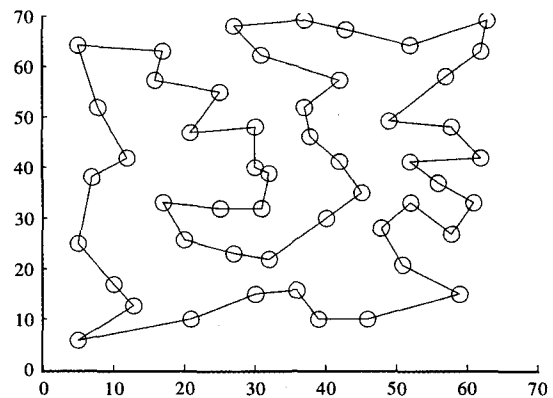


图 1 整数解最优路径

由表 2 可知, 在 50 次独立实验中, BAHA 有 12 次能收敛到最优解 426, 且最快可在第 52 代收敛, 算法最优解的平均值为 433.1; 与 ACS、BEGA 相比, BAHA 在搜索的精度和速度上都有了较大提升, 通过算法平均值的对比, BAHA 具有较好的稳定性; 且 BAHA 在保证搜索性能的情况下, 其稳定性优于文献[8, 10].

表 2 eil51 问题实验结果

算法	Lbest	NC	Nc	ns	La	Lb
文献[8]	430.7	/	/	/	446.3	
文献[10]	426	51	/	/	442.4	
ACS	432	50	0	299	439.4	426
BEGA	438	50	0	156	441.0	
BAHA	426	50	12	52	433.1	

采用浮点数方式求解 eil51 问题更符合实际情况,目前文献对其研究较少,因此,本文对浮点数方式求解 eil51 问题进行了仿真测试.浮点数表示城市间的距离时,其最优路径如图 2 所示.

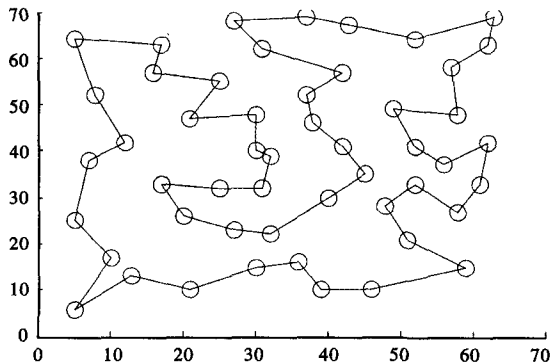


图 2 浮点数解最优路径

图 2 所示路径的浮点数解为 428.871 8,其整数解为 427.图 1 所示路径的浮点数解为 429.983 3,其整数解为 426;由此可知,当采用浮点数方式求解时,最优路径如图 2 所示,采用整数方式求解时,最优路径如图 1 所示.

以 6 吸嘴旋转式贴片机为原型,对实际生产中使用的 5 种不同元器件个数和种类的 PCB 板的贴装顺序进行优化计算,分别采用 ACS、BEGA 和 BAHA 独立运行 50 次,优化结果如表 3 所示.

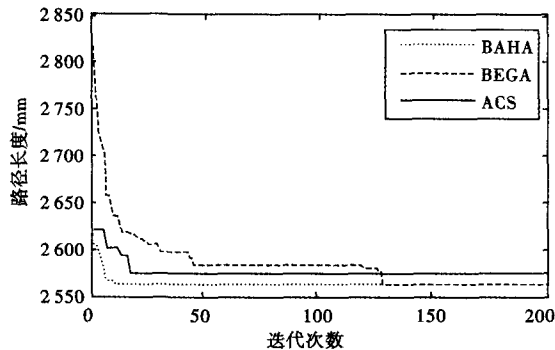
表 3 实例数据实验结果

标号	元件个数	元件种类	ACS/ mm	BEGA/ mm	BAHA/ mm
1	21	5	2 575.0	2 563.3	2 563.3
2	31	8	4 305.7	4 325.4	4 264.6
3	54	19	6 755.0	6 873.6	6 618.4
4	90	24	11 467	12 986	11 366
5	110	27	14 678	16 383	14 524

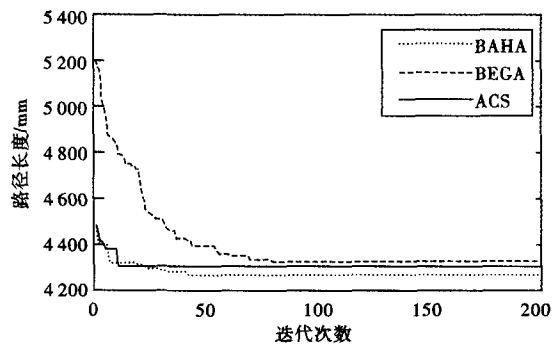
由表 3 可知,在元器件较少时,四种算法对不同

元器件个数和种类的 PCB 板的贴装顺序进行优化计算得到的最优解差距较小,但随着元件个数及种类的增加,BAHA 求得的最优解与其他算法的最优解差距逐渐增加.以第五个 PCB 板为例,当元件个数为 110 时,采用 BAHA 得到的路径与 ACS 和 BEGA 相比路径差距分别为 154 mm 和 1 859 mm.

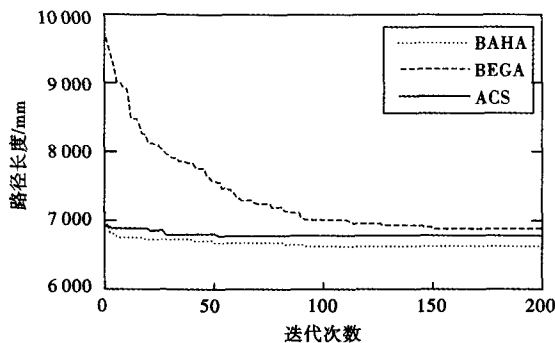
对 5 种 PCB 板贴装顺序优化计算的进化曲线如图 3 所示,图 3(a)-(c)分别对 BAHA、BEGA 和 ACS 三种算法得到的结果进行对比;对于第 4 和第 5 个 PCB 板,BAHA 在 200 代未得到最优解,且最终得到的解远大于其他两种算法得到的结果,所以图 3(d)-(e)只对 BAHA 和 ACS 两种算法的结果进行对比.



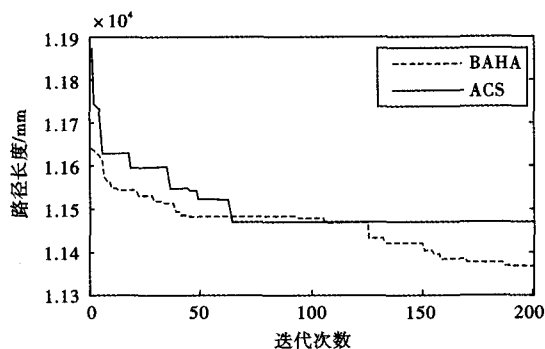
(a) PCB 板 1 进化曲线



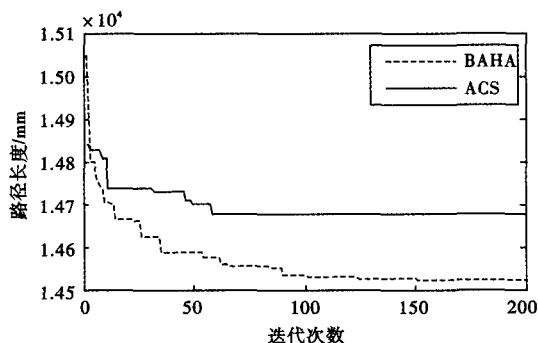
(b) PCB 板 2 进化曲线



(c) PCB 板 3 进化曲线



(d) PCB板4进化曲线



(e) PCB板5进化曲线

图3 5种PCB板贴装顺序优化的进化曲线

由图3(a)可知,元件个数为21,元件种类为5时,除ACS之外,其他两种算法均能收敛到最优解,其中BAHA在第10代收敛,收敛速度明显优于其他两个算法;由图3(b-e)可知,随着元件数目及种类逐渐增加时,BAHA求得的最优解与其他算法的最优解差距逐渐增加,如图3(e)所示,BAHA最优解远优于ACS;ACS在58代收敛于局部最优解,而BAHA有效地解决了早熟问题,能够有效地缩短贴装路程,提高贴装效率。

5 结束语

贴片机是PCB组装生产线中的瓶颈设备,BAHA通过两个种群的融合实现信息共享,提高算法的收敛速度和初期寻优能力;采用改进的OX的交叉算子,合理保留了优秀个体基因的排列顺序;加入局部搜索算子,在当代最优解附近进行更加精细的搜索;信息素重置防止陷入局部最优解,仿真结果表明BAHA收敛速度快,寻优能力强。通过对5种不同元器件个数和种类的PCB板的贴装顺序进行优化计算,结果表明,BAHA有较好的全局搜索能力

及求解能力,能够缩短贴装路程,提高贴装效率,有效缓解贴片机对整个生产线生产效率的影响。

参考文献:

- [1] 王君,罗家祥,胡跃明. 基于改进蚁群算法的贴片机贴装过程优化[J]. 计算机工程, 2011, 37(14): 256-258.
- [2] 闫红超,姜建国. 一种基于改进混合遗传算法的贴片机装配工艺优化方法[J]. 微电子学与计算机, 2006, 23(6): 213-215.
- [3] 陈铁梅,罗家祥,胡跃明. 基于蚁群-遗传算法的贴片机喂料器分配优化研究[J]. 控制与决策, 2011, 26(6): 929-932.
- [4] Kuo R J. Application of a hybrid of genetic algorithm and particle swarm optimization algorithm for order clustering[J]. Decision Support Systems, 2010, 49(4): 451-462.
- [5] Jiang Jianguo, Du Zhiping, Liu Chunyan, et al. Ant colony algorithms with characteristic of clustering for surface mount technology optimization[J]. IEEE transactions on Wireless Communications Networking and Mobile Computing, 2010;1-4.
- [6] 孟伟,韩学东,洪炳镨. 蜜蜂进化型遗传算法[J]. 电子学报, 2006(7): 1294-1300.
- [7] Dorigo M, Gambardella L M. Ant colony system: a co-operative learning approach to the traveling salesman problem[J]. IEEE transactions on Evolutionary Computation, 1997, 1(1): 53-66.
- [8] 顾军华,范培培,宋庆增,等. 改进的求解TSP问题文化蚁群优化方法[J]. 计算机工程与应用, 2010, 46(26): 49-52.
- [9] 刘勇,刘念,刘孙俊. 一种基于免疫蚁群混合算法的TSP求解模型[J]. 四川大学学报:工程科学版, 2010, 42(3): 121-126.
- [10] 饶卫振,金淳,黄英艺. 求解TSP问题的最近邻域与插入混合算法[J]. 工程理论与实践, 2011, 31(8): 1419-1428.

作者简介:

王超学 男,(1967—),副教授,博士. 研究方向为智能计算及其在电力系统中的应用。

孙有田 女,(1986—),硕士研究生. 研究方向为智能信息系统及其应用。

董慧 女,(1965—),副教授,博士研究生. 研究方向为智能信息系统及其应用。

崔杜武 男,(1945—),博士生导师,教授. 研究方向为电力电子与电力传动、计算机应用技术。