Research article

A tabu search heuristic for the component assignment problem in PCB assembly

Y.F. Wan and
P. Ji

The authors

Y.F. Wan is a Research Student and **P. Ji** is an Associate Professor, both in the Department of Manufacturing Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong.

Keywords

Assembly, Electronics industry, Printed circuits, Computer programming

Abstract

Discusses the component assignment problem in PCB assembly, where assigning components to appropriate machines, in order to get a minimum assembly time for the assembly line, can be formulated as an integer linear programming model. In order to obtain the optimal solution to the component assignment problem, the branch-and-bound method can be applied. However, it is not efficient. Applies the tabu search heuristic to the component assignment problem. The procedure of the tabu search to the problem is presented, and a numerical example is provided. Finally, the performance of the tabu search is analyzed with the example.

Electronic access

The research register for this journal is available at
http://www.mcbup.com/research_registers

The current issue and full text archive of this journal is available at
http://www.emerald-library.com/ft

Introduction

Printed circuit board (PCB) assembly is a prime manufacturing activity in many electronics companies. In general, an assembly line consists of multiple non-identical placement machines. The assignment of different components to various machines is critical, because it determines the efficiency and the productivity of the line, and it has attracted the attention of many researchers. Drezner and Nof studied a pick-and-insert sequencing problem for a robot in PCB assembly (1984). Ball and Magazine provided a framework with three basic levels for productivity improvement of a PCB assembly (1988). McGinnis *et al.* stated that a number of decisions have to be made from the interaction between process planning, production planning and scheduling for the production of a PCB assembly line (1992). Recently, Sze *et al.* presented a mathematical model for the component assignment problem in PCB assembly (2001).

There are numerous alternative methods to tabu search (TS) such as genetic algorithms, simulated annealing and artificial neural networks. However, there is no clear guidance available regarding the optimal choice for a given situation and further work is in progress in this area.

The problem to be studied in this paper, like many other problems in PCB assembly, can be formulated as an integer linear programming model, which is a typical discrete combinatorial problem. The traditional method to solve an integer linear model is the branch-and-bound (B&B) algorithm. The basic idea of the B&B method is to divide an integer linear programming problem into sub-problems, relax the integer requirement and solve the sub-problems as a linear programming model. This iteration is repeated until the optimal integer solution is found. However, the efficiency of the B&B method is very low in terms of computations since it needs to take quite a large number of iterations to find the optimal solution. So to be practical, some heuristic methods, like TS, have been devised. Although they cannot guarantee that the solution is optimal, they are efficient. TS, first proposed by

The authors would like to thank the Hong Kong Polytechnic University for their support for this project under program No. G-V737.

Assembly Automation
Volume 21 · Number 3 · 2001 · 236–240

Glover (1977), is an iterative and adaptive search procedure for solving discrete combinatorial optimization problems. Since then, it has successfully been applied to obtain optimal or sub-optimal solutions to many classic combinatorial problems, such as job-shop scheduling (Dell'Amico and Trubian, 1993), the quadratic assignment problem (Kapov, 1994), and the travelling salesman problem (Laporte *et al.*, 1996). The basic idea of TS is, for a cost minimization problem, to continue exploration without being confounded by a lack of improving moves or by falling back into a local optimum. In each iteration, an admissible move is applied to the initial or current solution. The move then transforms the solution into its neighbor with a smaller cost.

The component assignment problem in PCB assembly will be briefly discussed in this paper. Then it will be formulated into an integer linear programming model. To solve the model, the B&B method can be applied; however, the efficiency of the method is very low. Therefore this paper will apply TS to the component assignment problem in PCB assembly. The authors believe that this is the first time that the TS method has been applied to the component assignment problem. The procedure for applying the TS to the problem is presented, and a real-life numerical example provided by a computer manufacturer is evaluated.

2. Problem formulation

In general, a PCB has hundreds of components to be assembled on an assembly line. On the other hand, the assembly line consists of several non-identical component placement machines. The placement time for different component types varies with the placement machines. The component assignment problem is to assign the components to the placement machines in the line, so that a minimum cycle time can be achieved. Here, the cycle time is the maximum assembly time required by the machines, including machine setup time, and component placement time. So, if the line has m machines and the board has n types of components, the component assignment problem can be formulated as an integer linear programming model as follows[4]:

$$\text{Minimize } T \tag{1}$$

Subject to

$$T - \sum_{j=1}^{n} t_{ij}\, x_{ij} \geq s_i \text{ for } i = 1, 2, \ldots, m \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} = c_j \quad \text{for } j = 1, 2, \ldots, n \tag{3}$$

$$T \geq 0, \; x_y \geq 0 \text{ and integer}$$
$$\text{for } i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n \tag{4}$$

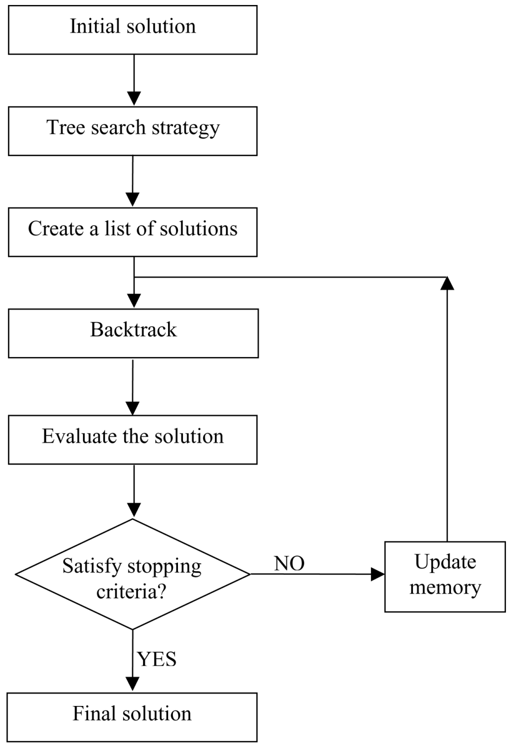
Here, T is the cycle time to be found, t_{ij} is the unit placement time for machine i and component j, x_{ij} (a decision variable) is the quantity of component j to be assembled by machine i, s_i is the set-up time required by machine i, while c_j is the total quantity requirement of component j in the board. The objective of the model is to minimize the cycle time T in the line. Constraint (2) states that the total assembly time (including machine set-up time and placement time) required by machine i cannot exceed the cycle time T, while constraint (3) is from the quantity requirement of component j in the board. Constraint (4) is obvious, since the quantity of a component to be assembled by a machine must be integer.

3. Tabu search

In general, the B&B algorithm can be adopted to solve an integer linear programming model. In model IP1, the number of decision variables for the component assignment problem is $mn + 1$. Consequently, the efficiency of the B&B algorithm will be very slow. So, this paper applies TS to the component assignment problem.

The overall structure of the TS heuristic for the component assignment problem is shown in Figure 1. At each iteration, that is, at each node of the search tree, an analysis is performed to identify which decision variable x_{ij} will be selected for immediate exploration, and this generates a more constrained linear programming model to be solved to continue the process, while the alternative branch is saved (the memory is updated) to be explored at a later date. The TS heuristic then randomly selects a decision variable to resume the search from early levels of the search tree.

Figure 1 Overall structure of TS for the component assignment problem



This TS heuristic has the following characteristics:

- The initial solution. At the beginning, model IP1 is solved as a linear programming model, that is, the integer requirement is relaxed, and its objective value acts as the lower bound for the TS. Then the fractional solutions are rounded to integer numbers with the new objective as the upper bound.
- The tree search strategy is used in the heuristic, like the B&B algorithm.
- In the backtracking moves, a branching variable is randomly selected. This is different from the B&B algorithm. In general, the B&B algorithm just backtracks to the previous level in the tree. This TS backtracks at a random level, depending on the chosen probability.
- At each node of the tree, a linear programming model is solved to generate a better solution.

4. Numerical example

A numerical example is presented to illustrate the efficiency of the TS heuristic. The data for the example with three placement machines and six types of components are shown in Table I.

The integer linear programming model for the numerical example is as follows:

$$\text{Minimize } T$$

Subject to

$$\begin{aligned}
&T - 3x_{11} - 7x_{12} - 7x_{13} \geq 110 \\
&T - 7x_{21} - 12x_{22} - 17x_{23} - 24x_{24} - 17x_{25} \\
&\quad - 24x_{26} \geq 147 \\
&T - 23x_{21} - 38x_{32} - 35x_{33} - 38x_{34} - 38x_{35} \\
&\quad - 36x_{36} \geq 147 \\
&x_{11} + x_{21} + x_{31} = 321 \\
&x_{12} + x_{22} + x_{32} = 67 \\
&x_{13} + x_{23} + x_{33} = 35 \\
&x_{14} + x_{24} + x_{34} = 12 \\
&x_{15} + x_{25} + x_{35} = 31 \\
&x_{16} + x_{26} + x_{36} = 12 \\
&T, x_{ij} \geq 0 \text{ and integer} \\
&\text{for } i = 1, 2, 3; j = 1, 2, \ldots, 6 \qquad \text{(IP2)}
\end{aligned}$$

As discussed earlier, the initial solution is based on the linear programming relaxation and it is obtained from the simplex method with x_{11} = 321, x_{12} = 1.51, x_{13} = 35, x_{22} = 65.49, x_{25} = 23.27, x_{34} = 12, x_{35} = 7.73, x_{36} = 12, and T = 1328.56.

The final result of the TS heuristic depends on the parameters preset, particularly, the probabilities for backtracking levels. For some probability sets (as shown in Table II) with 1,000 iterations performed (the stopping criteria), the best integer solution founded by the TS heuristic is T = 1333 with x_{11} = 319, x_{12} = 3, x_{13} = 35, x_{21} = 1, x_{22} = 64, x_{25} = 24, x_{31} = 1, x_{34} = 12, x_{35} = 7, x_{36} = 12, which happens to be the optimal solution to the original problem IP2. However, this is not guaranteed, since TS is a heuristic method.

5. Performance analysis

The performance of the TS was analyzed from two aspects:

(1) How much chance (or how many times) to obtain the optimal solution (quality).
(2) How many nodes backtracked (or how much time spent) to get the first optimal solution (quantity)?

The performance of the TS heuristic depends on the probabilities preset to determine which level of nodes is to be backtracked. Ten sets of

Assembly Automation
Volume 21 · Number 3 · 2001 · 236–240

Table I A numerical example

Machine i	Component type j 1	2	3	4	5	6	Set-up time S_i
1	3	7	7	∞	∞	∞	110
2	7	12	17	24	17	24	147
3	23	38	35	38	38	36	147
Number of component c_j	321	67	35	12	31	12	

probability ranges are studied for performance analysis, as shown in Table II.

For example, for set 1, if the randomly generated number is 0.3 at a node, then the TS will backtrack at two levels instead of one in the general B&B algorithm. For each probability set, 1,000 iterations were run and the performance of the TS is shown in Table III. For instance, for probability set 1, a solution with the cycle time of $T = 1336$ was obtained 310 times among the 1,000 iterations, that is 31 per cent. The best solution obtained for probability set 1 was $T = 1335$. Table III implies that the probability set 10 has over 50 per cent chance of getting the optimal solution of $T = 1333$, that is, probability set 10 is good in terms of quality. However, this probability set required the system to backtrack 174 nodes, that is, to solve a linear programming model with the simplex method 174 times, so it is not good in terms of quantity. On the other hand,

Table II Probability sets for backtracking levels

Set	Backtracking level: Level 1	Level 2	Level 3	Level 4
1	(0.000, 0.100)	(0.100, 0.400)	(0.400, 0.700)	(0.700, 1.000)
2	(0.000, 0.200)	(0.200, 0.467)	(0.467, 0.733)	(0.733, 1.000)
3	(0.000, 0.300)	(0.300, 0.533)	(0.533, 0.767)	(0.767, 1.000)
4	(0.000, 0.400)	(0.400, 0.600)	(0.600, 0.800)	(0.800, 1.000)
5	(0.000, 0.500)	(0.500, 0.667)	(0.667, 0.833)	(0.833, 1.000)
6	(0.000, 0.600)	(0.600, 0.733)	(0.733, 0.867)	(0.867, 1.000)
7	(0.000, 0.700)	(0.700, 0.800)	(0.800, 0.900)	(0.900, 1.000)
8	(0.000, 0.800)	(0.800, 0.867)	(0.867, 0.933)	(0.933, 1.000)
9	(0.000, 0.850)	(0.850, 0.900)	(0.900, 0.950)	(0.950, 1.000)
10	(0.000, 0.900)	(0.900, 0.933)	(0.933, 0.967)	(0.967, 1.000)

Table III Percentage: some feasible solutions obtained for different probability sets

T	Probability set 1 (%)	2 (%)	3 (%)	4 (%)	5 (%)	6 (%)	7 (%)	8 (%)	9 (%)	10 (%)
1333	0	0	1	3	5	8	15	28	40	54
1334	0	0	0	0	0	0	0	0	0	0
1335	1	3	6	11	14	20	26	32	31	31
1336	31	34	37	33	44	49	46	36	28	15
1337	0	0	0	0	0	0	0	0	0	0
1338	1	1	1	2	2	1	0	0	0	0
1339	41	40	34	37	23	14	9	3	1	0
1340	17	16	16	13	10	7	3	1	0	0
1341	0	0	0	0	0	0	0	0	0	0
1342	0	0	0	0	0	0	0	0	0	0
1343	3	3	2	1	1	0	1	0	0	0
1344	0	0	0	0	0	0	0	0	0	0
1345	3	2	1	1	1	0	0	0	0	0
1346	0	0	0	0	0	0	0	0	0	0
1347	3	2	1	1	1	0	0	0	0	0

probability set 3 required about 40 backtracking nodes, while set 5 was about 50 nodes. The reason behind this is that probability set 10 has a very high chance to backtrack only one level, very similar to the B&B algorithm, so it requires backtracking to more nodes to get the optimal solution, while probability set 3 has a higher chance to backtrack two, three, or even four levels, so it may obtain the optimal solution quickly. To compromise between the quality (the chance to get the optimal solution) and the quantity (the nodes backtracked), set 5 is recommended for the component assignment problem if the TS heuristic is used.

6. Conclusions

This paper presented a TS heuristic for the component assignment problem in order to obtain the optimal cycle time for a PCB assembly line. Compared with the B&B algorithm, the TS can backtrack to a different level randomly in the tree. Also a numerical example was provided. Finally, the performance of the TS heuristic with the different probabilities for backtracking was investigated in the paper.

References

Ball, M.O. and Magazine, M.J. (1988), "Sequencing of insertions in printed circuit board assembly", *Operations Research*, Vol. 36 No. 2, pp. 192-201.

Dell'Amico, M. and Trubian, M. (1993), "Applying tabu search to the job-shop scheduling problem", *Annals of Operations Research*, Vol. 41, pp. 269-76.

Drezner, Z. and Nof, S. (1984), "On optimizing bin picking and insertion plans for assembly robots", *IIE Transactions*, Vol. 16 No. 3, pp. 262-70.

Glover, F. (1997), "Heuristics for integer programming using surrogate constraints", *Decision Science*, Vol. 8, pp. 156-66.

Kapov, J.S. (1994), "Extensions of a tabu search adaptation to the quadratic assignment problem", *Computers & Operation Research*, Vol. 21 No. 8, pp. 855-65.

Laporte, G., Potvin, J.Y. and Quilleret, F. (1996), "A tabu search heuristic using genetic diversification for the clustered traveling salesman problem", *Journal of Heuristics*, Vol. 2, pp. 187-200.

McGinnis, L.F., Ammons, J.C., Carlyle, M., Cranmer, L., Depuy, G.W., Ellis, Y.P., Tovey, C.A. and Xu, H. (1992), "Automatic process planning for printed circuit card assembly", *IIE Transactions*, Vol. 24 No. 4, pp. 18-30.

Sze, M.T., Ji, P. and Lee, W.B. (2001), "Modeling the component assignment problem in PCB assembly", *Assembly Automation*, Vol. 21 No. 1, pp. 55-60.