

## Research Article

### PCB Assembly Process Optimization based on Constraint Model

Xuan Du, Yi Zhang and Lili Sun

College of Mechanical and Material Engineering of China Three Gorges University,  
Yichang 443002, China

**Abstract:** The structure and assembly process of Chip Shooter (CS) machine are studied. The factors of engineering practice such as the component size and machine structure etc. are considered. The assembly optimization model of Printed Circuit Board (PCB) on CS machine is formulated. A hybrid genetic algorithm (HGA) which combine with Polychromatic Sets (PS) and Genetic Algorithm (GA) is presented. A binary piece-wise coding method is proposed. In the individual chromosome, feeder arrangement and component placement sequence is described simultaneously. Constraint model is used to consider the impact of component size on the assembly speed and the feeder arrangement of CS machine. Meanwhile the constraint model improved the quality of initial population, ensured the effectiveness of GA operation and narrowed the search space of GA obviously. Calculation example indicated that HGA optimize the component sequence and feeder arrangement simultaneously and achieve excellent practicability. The assembly efficiency of CS machine is improved obviously.

**Keywords:** Constraint model, chip shooter machine, PCB assembly optimization, polychromatic sets

#### INTRODUCTION

The Chip Shooter (CS) machine is a kind of high speed assembly machine which used in the printed circuit board (PCB) assembly process with SMT. In order to improve the assembly efficiency of PCB, the PCB assembly optimization caused more and more attention (Leu *et al.*, 1993; Ong and Khoo, 2002; Deo *et al.*, 2002; Wilhelm and Tarmy, 2003; Zeng, 2011). PCB assembly optimization of the CS machine is too complex. Although many researchers are able to solve two NP optimization problems (i.e., the component sequence and feeder arrangement problem) simultaneously. Restrained by complexity of the PCB assembly optimization problems, it is simplified and the impact of engineering practice (i.e., component size etc.) can not be considered (William and Ji, 2007; William and Ji, 2003; Du *et al.*, 2008). Actually, component size has great impact on the assembly process and speed of CS machine. It can not be ignored. Based on contour matrix of polychromatic sets (PS) (Pavlov, 2006), the constraint model of PCB assembly process is presented. It combine with Genetic Algorithm (GA) to achieve the PCB assembly optimization. Not only the component sequence and feeder arrangement are optimized simultaneously, but also the component size and strategy of feeder arrangement are considered. HGA has excellent calculation efficiency and can improve the PCB assembly efficiency. The method is presented in this

research provides a new method to solve complex engineering optimization problems.

CS machine has a feeder carrier loading component, a moveable PCB table carrying PCB and a rotary turret with multiple assembly heads. The feeder carrier that loads many feeders can move in x-direction. PCB table can move in X-Y directions. Each assembly head has several nozzles with different sizes in order to pick up and place components with different size. Three mechanisms of chip-shooter machine move simultaneously. The movement time of PCB table is used to carry PCB from one position to next position. The movement time of feeder carrier is used to pick component from feeder. The rotary time of turret is used to rotate one step to pick and place component.

During the assembly process, the operating sequence of the CS machine is described as follows. As the first PCB of a batch enters the machine, the first nozzle of the turret picks up a component from a feeder. Then the turret indexes one step and the next nozzle picks up the second component. After that, the turret indexes again to pick up the next component and so on. At the same moment, the PCB table is moved to the placement location waiting for the first component to be placed on the PCB. When the seventh component is being picked up, if the turret has 12 heads, the first component is being placed on the PCB at the same time. These operations continue so that the turret indexes one step, the feeder carrier moves to the location containing the next pickup component and the

**Corresponding Author:** Xuan Du, College of Mechanical and Material Engineering of China Three Gorges University, Yichang 443002, China, Tel.: 13972605940

This work is licensed under a Creative Commons Attribution 4.0 International License (URL: <http://creativecommons.org/licenses/by/4.0/>).

PCB table moves to the next placement location. In the assembly of the last six components, there is no need to pick up components for the PCB being assembled. However, the nozzles of the turret can pick up the first six components for the next board to be assembled, if necessary. For the first few components assembled in a batch of PCBs, there are only pickup movements and no placement movement. For the last few components of the same batch, there are only placement movements and no pickup movement (William and Ji, 2007).

In this study, we formulated an integrated optimizing model to describe the component sequence and feeder arrangement simultaneously and proposed an effective algorithm to optimize PCB assembly optimization problem of CS machine.

### OPTIMIZATION MODEL

During the assembly process of CS machine, component is picked up and placed simultaneously. The assembly time is dependent on the movement time of three parallel moveable mechanisms: the movement time of PCB table, the movement time of feeder carrier and rotary time of turret. The longest time among them is the assembly time of a component. The objective of the assembly optimization for CS machine is to reduce the assembly time of PCB to improve the assembly efficiency.

**The movement time of PCB table:** PCB table can move in X and Y directions simultaneously.  $t_t(s)$  is the time used by the PCB table for traveling from the location of component (s-1) to the location of component s:

$$t_t(s) = \max \left\{ \frac{|x_s - x_{s-1}|}{V_x s_s}, \frac{|y_s - y_{s-1}|}{V_y s_s} \right\} \quad s=1 \text{ to } N \quad (1)$$

where,

- $x_s$  and  $x_{s-1}$  = The x-coordinates of component s and s-1, respectively
- $y_s$  and  $y_{s-1}$  = The y-coordinates of component s and s-1, respectively
- $V_x$  and  $V_y$  = The speeds of PCB table in the x and y directions, respectively

$s_s$  is the slowest speed factor of PCB table when sth component is assembled on PCB. It describes the effect of different components size on assembly speed.

During the assembly process, the speed of PCB table is changeable to guarantee the assembly quality. The speed of PCB table is decided by the biggest size component that has been assembled on PCB. It means that the speed of PCB table can only be reduced and not be raised during the assembly process of a PCB. So the distance between the components has great effect on the assembly time as well as the assembly sequence of different component size.

**The movement time of feeder carrier:** The feeder carrier on CS machine moves from position of feeder which loads component s-1 to the position of feeder which loads component s. the movement time of feeder carrier (i.e.,  $t_f(s)$ ) can be calculated as follow:

$$t_f(s) = |F_s - F_{s-1}| t_s \quad (2)$$

where,  $t_s$  is a unchangeable movement time of feeder carrier which move a slot.

On the feeder carrier,  $F_s$  and  $F_{s-1}$  are the positions of feeder which load component s and s-1, respectively.

On the feeder carrier, the wide of a slot is fixed, but the size of components is different. So the feeder which load component with small size occupy one slot and the feeder which load component with big size occupy more than one slot on the feeder carrier. The number of slot between the adjacent feeders on the feeder carrier may be different. Eq. (2) considers the difference of feeder size and can calculate the movement time of feeder carrier accurately.

Furthermore, the strategy of feeder arrangement can affect the assembly process obviously. In order to improve the efficiency, if the assembly number of a component type is more than other component type, the feeder load this component type will be arranged twice or more. During the assembly process, a suitable feeder must be selected from multiple feeders which load same component type.

For example, the positions of feeder which load component s-1, s and s+1 are  $F(s-1)$ ,  $F(s)$  and  $F(s+1)$ , respectively. If the component type s is loaded on two feeders. The positions of them are  $F_1(s)$  and  $F_2(s)$ , respectively. If

$$\begin{aligned} & |F(s-1) - F_1(s)| + |F_1(s) - F(s-1)| \\ & < |F(s-1) - F_2(s)| + |F_2(s) - F(s-1)| \end{aligned}$$

the component s is picked from feeder  $F_1(s)$ , otherwise the component s is picked from feeder  $F_2(s)$ .

**Rotation time of turret:** Assume there are M assembly heads on the turret of CS machine, the turret rotation time  $t_r(s)$  is used by the turret for rotating one step. The component size affects the rotary speed of turret. In order to assure the operation reliability, the rotary speed of turret will decrease when the assembly head picks up the component with bigger size. The rotary speed is decided by the picked component whose size is the biggest in the M/2 components on the turret.  $t_r(s)$  can be calculated as follows:

$$t_r(s) = \max_{s \leq m \leq M/2} \left\{ \frac{tr}{r_m} \right\} \quad (3)$$

where,  $tr$  is the standard time which is used by the turret for rotating one step.  $r_m$  is the speed factor of mth component which is picked on the turret.

**The optimization model of PCB assembly time:** The longest one among the three times in one step is called the dominating time. So, for the CS machine, the objective of the assembly optimization is obviously to minimize the total assembly time, which is the summation of all dominating times of components so that the highest productivity of the CS machine can be achieved. The optimization model of PCB assembly on CS machine can be formulated as follows:

$$\min T = \left. \begin{aligned} & \sum_{i=1}^N \sum_{j=1}^N t_i(s) x_{i(s-1)} x_{js} \\ & \max_{s=1}^N \left\{ \sum_{i=1}^N \sum_{j=1}^N \sum_{t=1}^P \sum_{h=1}^L \sum_{k=1}^L t_f(s) x_{i(s+M/2)} x_{j(s+M/2+1)} x_{tih} x_{tjk} \right. \\ & \left. t_r(s) \right\} \end{aligned} \right\} \quad (4)$$

s.t.

$$\sum_{i=1}^N x_{is} = 1 \quad (5)$$

$$\sum_{s=1}^N x_{is} = 1 \quad (6)$$

$$\sum_{h=1}^L x_{tih} \geq 1 \quad (7)$$

$$\sum_{k=1}^N x_{tjk} \geq 1 \quad (8)$$

$$i \neq j, \quad P \leq L$$

$$i=1,2,\dots,N; \quad j=1,2,\dots,N; \quad s=1,2,\dots,N$$

$$t=1,2,\dots,P; \quad h=1,2,\dots,L; \quad k=1,2,\dots,N$$

where,

$M$  = Number of assembly head on the turret

$L$  = Number of slots on the feeder carrier

$P$  = Number of component types

$i, j$  = Indices of component

$t$  = Indices of component type

$s$  = Indices of assembly sequence of component

$x_{is}$  = 0-1 variable, 1 if the placement sequence of component  $i$  is  $s$  and 0 otherwise

$x_{tih}$  = 0-1 variable, 1 if the type of component  $i$  is  $t$  which is uploaded in slot  $h$  and 0 otherwise

Eq. (5) and (6) are to guarantee that exactly one component is only placed in one position. Equation (7) and (8) mean a feeder will be uploaded more than once on the feeder carrier. The component size is considered, a feeder which upload bigger component may be placed in more than one slot. So  $P \leq L$ . During the

practical assembly process,  $M/2$  components have been picked up on the turret when the first component is placed in the position. The difference between placement sequence and pickup sequence is described in the model. The  $M/2$  components have been picked on the turret, when the first component is placed in the position.

## HYBRID ALGORITHM

During the assembly process of CS machine, the component placement sequence and feeder arrangement influence each other. The strategy and component size make it more different to optimize. To examine the complexity of the models, it is essential to find the numbers of variables and constraints in the model, in Eq. (4) to (8), it can be seen that the model is very sophisticated. The objective function is nonlinear and also its enumeration is huge. In order to solve it efficiently, GA has been combined with the constraint model which is formulated by PS contour matrix. The flow chart of algorithm is shown in Fig. 1.

### PS model of the assembly component:

- **Component define:** Component size has great effect on the assembly speed of the CS machine in the assembly process. The component sequence and feeder arrangement should consider the component size. In this research, according to the component size, the components are classified as key component and free component on a PCB.

The index of key component is identical to the assembly sequence of it. Different key components have different assembly speed. On a PCB, the number of key component is the biggest among the components with same assembly speed.

- **The description of assembly component based on PS:** In this research, there are five speeds to assemble all the component with different size. Figure 1 describes the component based on PS contour matrix.  $F_1-F_5$  correspond with five different speeds, whose factor is from one to five.  $a_i$  describe the components on PCB. In the contour matrix, the element value represent the specific speed factor, 6 is used to describe the key component.  $F(a_i)$  describe the classification and speed factor of a specific component. In the Table 1, component  $a_1, a_2, a_6, a_9, a_{14}$  are key component. For example,  $F(a_1) = (60000)$  indicate component  $a_1$  is key component. Its speed factor is 1.  $F(a_{10}) = (00300)$  means component  $a_{10}$  is free component, its speed factor is 3.
- **Relation description of the assembly component:** Based on PS contour matrix, not only the speed factors of different components are

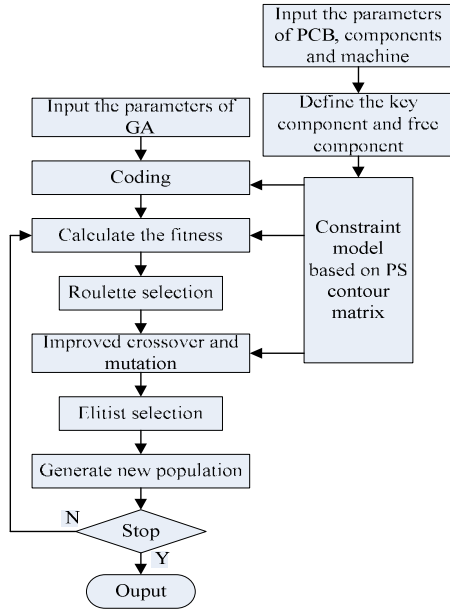


Fig. 1: Flow chart of algorithm

Table 1: PS contour matrix sketch map of assembly component

	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$
$a_1$	6	0	0	0	0
$a_2$	0	0	0	6	0
$a_3$	1	0	0	0	0
$a_4$	1	0	0	0	0
$a_5$	0	2	0	0	0
$a_6$	0	0	6	0	0
$a_7$	0	2	0	0	0
$a_8$	0	2	0	0	0
$a_9$	0	6	0	0	0
$a_{10}$	0	0	3	0	0
$a_{11}$	0	0	0	4	0
$a_{12}$	0	0	0	4	0
$a_{13}$	0	0	0	0	5
$a_{14}$	0	0	0	0	6
$a_{15}$	0	0	3	0	0

described, but also the relations among different components are indicated. The relation can be classified as follows:

- Component  $a_i$  and  $a_j$  have the same speed factor,  $AF_1(a_i, a_j) = 1$
- The speed factor are different, but the difference is in a range,  $AF_2(a_i, a_j) = 1$
- The speed factor are different and the difference is big,  $AF_3(a_i, a_j) = 1$

Table 2: Chromosome structure

Component type	Component type														
1	1	1	2	2	3	4	5	6	1	2	3	4	5	6	1
1	2	3	4	5	6	7	8	9	1	2	3	5	8	10	11
Indices of components	Indices of slots														

Table 3: Chromosome initialization of component sequence part

Gene site	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Indices of component	$a_1$	$a_3$	$a_9$	$a_4$	$a_8$	$a_5$	$a_6$	$a_7$	$a_{10}$	$a_2$	$a_{15}$	$a_{12}$	$a_{11}$	$a_{14}$	$a_{13}$
Speed factor	$F_1$	1	$F_2$	1	2	2	$F_3$	2	3	$F_4$	3	4	4	$F_5$	5

Based on the description of component relation, PS constraint model is formulated by the description of component relation. Combined with PS constraint model and GA, the hybrid algorithm can ensure the effectiveness and quality of the chromosome. So the search space of GA can be decreased obviously, the efficiency of algorithm is improved.

**Encoding and decoding:** Encoding and decoding is the key of GA to solve the problem. For PCB assembly optimization of CS machine, because the component sequence and the feeder arrangement problems are considered simultaneously, a chromosome should include two problems representations. The component sequence can be indicated by the indices of components. The feeder arrangement can be indicated by the slot position of feeder on the feeder carrier. Every component correspond a component type and a feeder upload the component with same component type. So the component sequence and feeder arrangement can be related by the component type in a chromosome. In this research, a binary piece-wise coding method is proposed. The chromosome structure is shown in Table 2.

Chromosome is made up of two parts, one shows component sequence and the other shows feed arrangement. In Table 3, consider a PCB with 9 components of six types. The component sequence starts with component 1, the components 2,3,4,5,6,7,8 and finally component 9, its sequence can be represented as (1 2 3 4 5 6 7 8 9). For the feeder arrangement is presented by (1 2 3 5 8 10 11), it means that two feeders which upload component type 1 is placed in slot 1 and slot 11 respectively on the feeder carrier, the component type 4 is placed in slot 5, component type 5 is placed in slot 8 and the component type 6 is placed in slot 10 and so on. The indices of slots are helpful to present the component size. A feeder upload bigger component can occupy more than one slot.

In a chromosome, all the gene values are represented by integer. The coordinates of a component can be obtained by the index of component. Combine with coordinates and speed factors of different components, the movement time of PCB table can be calculated. Movement time of feeder carrier can be calculated by indices of different slots of different feeders. The relation between component sequence

and feeder arrangement is represented by component type. It is helpful to solve them simultaneously.

**Population initialization based on constraint model:**

The component size will affect the assembly speed. During the assembly process of a PCB, the assembly speed of PCB table can not increase. So the smaller component with faster assembly speed is superior to bigger component, which is helpful to reduce PCB assembly time.

PS contour matrix describes component classification and the relations and it can be used to enhance quality of initial population and improve calculation efficiency of fitness. The generation of initial population is listed as follows:

- Step 1:** In a chromosome, the length of component sequence part is decided by the number of assembly component. As illustrated in Table 3, the length of component sequence part is 15.
- Step 2:** According to the number of key components, five gene site are selected(1,3,7,10,14). The key components are placed to corresponding gene site on descending sequence of speed.
- Step 3:** Searching PS model of component, the free component is placed on vacancy of chromosome on descending sequence of speed. In the chromosome, the part of feeder arrangement is generated randomly.
- Step 4:** Checking the relation between key component and adjacent component with PS constraint model. if  $AF_3(a_i, a_j) = 1$ , the chromosome will be abandoned to ensure the individual quality.
- Step 5:** Repeating steps above, until the initial population is generated.

**Genetic operation:** Efficiency of GA search is a bottleneck when GA is used to solve complex combinational optimization problems. It is easy to produce many chromosomes with poor quality and invalid solutions. In this research, genetic operation combine with PS constraint model to ensure the quality of chromosome, the efficiency of GA search can be improved.

- **Crossover based on constraint model:** During the assembly process of CS machine, the traditional crossover methods could damage chromosome and generate invalid solution. In order to ensure the quality of chromosome and improve the efficiency of research, a new crossover method is designed on the basis of PS constraint model. The crossover steps are illustrated as follows:

**Step 1:** Threshold "T" is determined. It means the biggest speed difference of adjacent component in the component sequence. The bigger of T means the bigger speed difference,

the search space of GA is bigger and the efficiency of search reduces.

- Step 2:** Generating a binary string whose length is identical to the chromosome. An element value corresponds to a gene of a chromosome.
- Step 3:** In the first parent chromosome, the gene which correspond to 1 of binary string is preserved and is copied to the first child chromosome. The values of genes which correspond to 1 of binary string form a list according to the order in the second parent chromosome.
- Step 4:** In the second parent chromosome, the gene which correspond to 0 of binary string is preserved and is copied to the second child chromosome. The values of genes which correspond to 1 of binary string form a list according to the order in the first parent chromosome.
- Step 5:** The gene list which is generated in step3 is placed in the vacant of the first child chromosome in order.
- Step 6:** The gene list which is generated in step 4 is placed in the vacant of the second child chromosome in order.
- Step 7:** Based on the  $F(a_i)$  of PS contour matrix, check the relations of adjacent component of child chromosome. If  $AF_1(a_i, a_j) = 1$  and  $AF_2(a_i, a_j) = 1$ , this chromosome is preserved, otherwise this crossover operation is invalid.
- Step 8:** Repeating steps 2 to 7, until all the crossover operations are completed.

The crossover operation which is proposed in this research ensures the index of component is not repeated. The feeder arrangement part ensures the number of feeders of parent chromosome and child chromosome is identical. A chromosome corresponds to an effective component assembly sequence and feeder arrangement. Furthermore, PS constraint model consider the speed relation between different components, the quality of chromosome can be ensured.

- **Self-adaption mutation based on constraint model:** Mutation can keep the diversity of GA population and avoid the premature convergence. In this research, mutation is combined with the PS constraint model to ensure the effectiveness and the quality of chromosome. The steps of mutation operation are listed as follows:

- Step 1:** A threshold "T" of speed factor is determined.
- Step 2:** Choose two gene sites randomly in one chromosome. It means two components are selected in the component sequence.
- Step 3:** Based on the  $F(a_i)$  of two components in the PS constraint mode, the speed of components are determined.
- Step 4:** Checking the relation between two components, if  $AF_1(a_i, a_j) = 1$  or  $AF_2(a_i, a_j) =$

1, two component can exchange their position in the chromosome, otherwise the mutation is invalid and two gene sites are choose again.

Along with the progress in the optimization search of GA, the differences among chromosomes are decreased gradually. At this time, increased mutation rate can keep search result away from local optimum and arrive at global optimization eventually. Thus the adaptive mutation rate is adopted. According to the difference of fitness value between adjacent generations, the mutation rate value is adjusted.

- **Fitness calculation based on PS model:** In this research, a chromosome describes the component assembly sequence and feeder arrangement simultaneously. With the combination of PS model and GA, not only the movement distance but also the relation and speed factor of components can be considered. The assembly time is calculated accurately and efficiently.

## COMPUTATIONAL RESULT AND ANALYSIS

Each chromosome contains two parts, the first part representing the sequence of component assembly, the second part indicate the feeder arrangement. So the PCB assembly optimization of CS machine can be solved by improved GA (IGA) which is proposed in this research. The calculation example which is provided in literature (Leu *et al.*, 1993) is calculated by improved IGA. The optimization result is 33.58s. The comparison between the results obtained from IGA and those obtained from other researchers show that IGA is better than many algorithms (William and Ji, 2007; Chen and Chyu, 2003). Furthermore, the optimization results are improved obviously when the feeders which upload component type 4, 9, 10 are placed twice on the feeder carrier. The optimization result is 23.5s.

For a PCB assembly case, without consider the component size, the movement distance of PCB table is 1040 mm, assembly time is 11.08s. The movement distance of PCB table which consider the component size, is 3871mm, the assembly time is 5.18s. It can be shown that the optimization results are more practical when the different assembly speeds of different components are considered.

Based on PS constraint model, the initial population quality is improved and the search space of GA is reduced. So the efficiency of IGA is improved obviously. In literature (William and Ji, 2007), the optimization result of the optimization problem of PCB assembly with 8 components and 8 feeders is obtained in 5s. In this research, the optimization result of the optimization problem of PCB assembly with 50 components and 13 feeders is obtained in 5s. The complexity of optimization problem increase obviously and the calculation time can not increase because of the PS constraint model.

## CONCLUSION

The lack of efficient optimization algorithm in the literature related to the practical PCB assembly optimization for CS machine is the main driving force behind this research. The movement mechanism of CS machine and the assembly process are analyzed in this study. An integrated model which describes the component sequence, feeder arrangement and considers engineering practices (i.e., assembly speed, component size and the strategy of feeder arrangement) is formulated. An effective algorithm is presented to optimize PCB assembly optimization problem of CS machine with the combination of PS constraint model and GA. The quality of initial population is improved and the search space is narrowed on the basis of PS constraint model, which lead to improve the calculation efficiency of algorithm.

## ACKNOWLEDGMENT

This study was supported by NSFC 51275274, Science Research Fund of Hubei Provincial Department of Education D20121306 and Science research fund of CTGU KJ2010B020.

## REFERENCES

- Chen, W.S. and C.C. Chyu, 2003. A Hybrid Genetic Algorithm for Solving Feeder Arrangement and Placement Sequencing Decisions in PCB Assembly [EB/OL]. Retrieved form: [Http:// machinevision.iem.yzu.edu.tw/pcb2002/paper/A2.pdf](http://machinevision.iem.yzu.edu.tw/pcb2002/paper/A2.pdf).
- Deo, S., R. Javadpour and G.M. Knapp, 2002. Multiple setup PCB assembly planning using genetic algorithms. *Comput. Ind. Eng.*, 42(1): 1-16.
- Du, X., Z.B. Li, *et al.*, 2008. Chip shooter machine pasting and assembling process optimization based on GA. *Xi'an Jiaotong Univ. J.*, 42(3): 295-299.
- Leu, M.C., H. Wong and Z. Ji, 1993. Planning of component placement/insertion sequence and feeder setup in PCB assembly using genetic algorithm. *J. Electron. Packaging*, 115(4): 424-432.
- Ong, N.S. and L.P. Khoo, 2002. Sequence placement planning for high-speed PCB assembly machine. *Integrat. Manuf. Sys.*, 13(1): 35-46.
- Pavlov, V.V., 2006. *Structural Simulation in CALS Technology*. Science Press, Moscow.
- William, H. and P. Ji, 2003. Component scheduling for chip shooter machines: A hybrid genetic algorithm approach. *Comput. Oper. Res.*, 30(14): 2175-2189.
- William, H. and P. Ji, 2007. *Optimal Production Planning for PCB Assembly*. Springer, London.
- Wilhelm, W.E. and P.K. Tarmy, 2003. Circuit card assembly on tandem turret-type placement machines. *IIE Trans.*, 35(7): 627-645.
- Zeng, Y., 2011. A workshop scheduling problem study based on IGA. *Mech. Design Manufacture*, 7: 180-182.