

AN APPROACH FOR BALANCING WORKLOAD IN PRINTED CIRCUIT BOARD ASSEMBLY

Patcharaporn Neammanee¹⁾, Sabah U. Randhawa²⁾

¹⁾ National Institute of Development Administration, School of Applied Statistics
Serithai Rd., Bangkok, 10240, Thailand (patchara@as.nida.ac.th)

²⁾ Office of Academic Affairs, Oregon State University, Corvallis,
Oregon, 97331, USA. (Sabah.Randhawa@orst.edu)

Abstract

The paper describes an approach to assign components to machines. A component assignment algorithm is developed to balance workload among machines within a production line. This workload includes the feeder setup time and the placement time. Classification of setup components into three categories: standard, semi-standard, and custom setup components, is performed to reduce the complexity of component assignment problem. The maximum difference between optimal and heuristic solutions is 10.8%, with the minimum difference being 4.5%. The approach is efficient in term of computation time and can be applied to industry problems.

In addition, impacts of two main parameters, which are threshold value and size of feeder capacity, are explored using the industry data. The result shows that size of feeder capacity, a key parameter for family grouping, would affect the total imbalance as a performance measure while the threshold value does not statistically influence total imbalance.

1. Introduction

With continuous advancements in the development of electronic components and automatic assembly machines, the manufacturing of Printed Circuit Boards (PCBs) has become more complex. This has resulted in seeking better process planning approaches to increase the throughput of PCB assembly. Component allocation is a major issue for process planning since it deals with a large number of components. To avoid one machine becoming a bottleneck and slowing down the entire line, the time for a PCB spent on each machine should be approximately equal in a coupled machine system. [1] addressed component allocation on two-coupled machines in order to maximize the production rate. The objective function of their Integer Programming (IP) model was to balance the placement time on the two machines. However, the component setup time was not taken into consideration.

In low-volume high-mix product environment, the feeder setup time becomes a critical issue in board assembly [2, 3, 4]. [4] reported that the changeover time including feeder setup and board setup time may occupy more than 35 percent of total machine time. The goal of component allocation problem in a coupled machine system is not only to maximize the balanced placement time across machines but also to minimize the setup time. [5, 6] presented an IP model and two heuristic models of component assignment to balance a combination of the placement time and the component setup time for each PCB type.

In reality, to save component setups, similar PCB types are grouped into families. All required components for a family might not be allocated on machines at the same time. Component allocation problem is related to setup strategies. For a multiple setup strategy, some components have to be loaded/unloaded more than once because of limited feeder capacity. [5, 7] determined two types of setup components: common components for all PCB types in a family called *standard* setup components and remaining components temporarily stored on feeders for one or a few board types called *custom* setup components. A standard setup component will be loaded only one time while the custom will be loaded several times to produce a PCB family. [8] divided the feeder bays into three categories: fixed, semi-fixed, and configurable feeder bays. A component in a fixed feeder bay was used to produce all PCB types. A component in the semi-fixed bay would be installed for the production of some PCB types. Finally, components in the configurable bay, the remaining set of components, varied from one board to the next.

Process planning in PCB assembly usually deals with many issues, such as board grouping, board sequencing, component allocation to machines, feeder assignment and placement sequencing. Solving one may impact to another. Several researches required some issues are already executed before seeking answers of the others or integrated these issues together. [9] combined component allocation with PCB sequencing and feeder assignment problem for two decoupled machines. Their research divided the problem into three sub-problems: component allocation, PCB sequencing, and feeder assignment. One of the developed heuristics in [10] focused on grouping all PCBs into

subfamilies based on total board dissimilarities at the component level and then components were assigned to machines with regard to an overall machine workload balancing constraint. [11] proposed scheduling algorithms, which is divided into two major phases: machine allocation and sequencing phases.

2. Methodology

In PCB assembly, there are a large number of different components required by all PCBs during a planning horizon. Setup strategies are used to determine the manner to group these boards until they can be produced with only one machine setup. Grouping similar PC boards has been used in many researches [3, 12, 13]. However, a board family may require a number of feeders that may exceed the feeder capacity, which means that producing such a family needs more than one machine setup [14]. One of these setup strategies is Decompose And Sequence (DAS), which divides a PCB family into subfamilies until the number of component feeders per subfamily is no more than feeder capacity [15]. The number of feeder setups is typically reduced if similar PC boards are produced next to each other [16]. Consequently, subfamilies sequencing is applied to determine board sequences within each family. *Keep Tool Needed Soonest* (KTNS) is a supplementary tool for board sequencing to keep some components on the free feeders in case the number of required components is less than feeder capacity [3, 17, 18]. Board grouping, family decomposition, subfamily sequencing, and KTNS procedures were addressed in several published articles. Examples include: [19, 20]; Best First Search, and [21].

The information required for component assignment procedure is a subfamily-to-component matrix for a family and a sequence of subfamilies. Components are then categorized into standard, semi-standard and custom setup components. A standard setup component is permanently installed on a machine during production of an entire family. A semi-standard setup component is allowed to stay on a machine to produce some subfamilies. Finally, a custom setup component is loaded/unloaded for an individual PCB subfamily. At this point, each component is defined as standard, semi-standard, or custom setup component. The allocation model and the solution approach are discussed with some assumptions in the following sections.

2.1 Assumptions

The following assumptions are made to keep the problem within manageable limits:

1. A number of slots to load all components required by a subfamily is not more than feeder capacity of a production line.
2. A component feeder can occupy more than one slot. A component can also be mounted on any slot of any machine.
3. A component type can be loaded multiple times on the same machine in a line. However, it can not be duplicated on particular machines at the same time.
4. All PCBs are grouped into families and subfamilies using Group Technology (GT) concept and feeder capacity limitation. Subfamily sequences are determined.
5. Each PCB is loaded only one time and splitting of PCB lots, one PCB type for each lot, is not allowed.
6. Feeder installation on all machines in a line will be done at the same time.

2.2 Mathematical Model

The objective is to balance workload across machines for individual PCB type. A machine's workload consists of component placement time and feeder setup time for all required components. This objective is equivalent to minimizing the sum of placement times and feeder setup times for all PCB types in a family.

Variables :

- i, j, m, g : index for component types, PCB types, machines, and PCB subfamilies, respectively
 q_j : volume of each PCB type j to be produced.
 d_{ij} : number of components i required by one board of PCB type j .
 t_{ijm} : processing time of picking and placing component type i on board type j using machine m .
 s_{im} : estimated setup time for component i on a feeder on machine m .
 f_i : number of slot required by component type i
 C_m : number of slots available on machine m
 N_g : set of components required by PCB subfamily g
 W_{igm} : number of setups for the semi-standard setup component i for subfamily g on machine m
 λ_{jg} : component placement time for each PCB type j in subfamily g
 α : feeder setup time for standard setup components
 Y_g : feeder setup time for semi-standard and custom components for subfamily g

Decision variables :

- $X_{im} = 1$ if component i is a standard setup component on machine m ,
 $= 0$ otherwise
 $Y_{igm} = 1$ if component i is a semi-standard setup component for PCB subfamily g on machine m ,
 $= 0$ otherwise
 $Z_{igm} = 1$ if component i is a custom setup component for PCB subfamily g on machine m ,
 $= 0$ otherwise

= 0 otherwise

Objective function :

$$\text{Min } \alpha + \sum_g (Y_g + \sum_j \lambda_{jg})$$

Constraints :

$$\lambda_{jg} \geq \sum_i q_j t_{ijm} d_{ij} \quad \forall i \in N_g, \forall j, m \quad (1)$$

$$Y_g \geq \sum_i s_{im} (W_{igm} + Z_{igm}) \quad \forall i \in N_g, \forall g, m \quad (2)$$

$$\alpha \geq \sum_i s_{im} X_{im} \quad \forall i, m \quad (3)$$

$$Y_{igm} - Y_{ig-1m} \geq W_{igm} \quad \forall i \in N_g, \forall g, m \quad (4)$$

$$\sum_i f_i (X_{im} + Y_{igm} + Z_{igm}) \leq C_m \quad \forall i \in N_g, \forall g, m \quad (5)$$

$$X_{im} + Y_{igm} + Z_{igm} \geq 1 \quad \forall i \in N_g, \forall g, m \quad (6)$$

$$X_{im}, Y_{igm}, Z_{igm}, W_{igm} \in \{0, 1\} \quad \forall i, g, m \quad (7)$$

The objective function is to minimize the sum of setup and placement time for board types in a family. Constraints (1), (2) and (3) determine processing time for producing a lot of each PCB type and component setup time for a PCB family. Constraint (4) represents the number of component setups for semi-standard components. Constraint (5) corresponds to feeder capacity constraint for each machine. Constraint (6) ensures that a component on a PCB would be exactly one type of setup component. Constraint (7) is the integrality requirements for decision variables.

2.3 Solution approach

The IP model for the type of problem, formulated in previous section, has been shown to be NP-complete. This implies that solving for optimality may not be feasible in terms of computation time. A heuristic approach was developed to obtain acceptable solutions in a reasonable amount of CPU time.

As mentioned earlier, three types of setup components are defined: Standard setup components based on the family level, semi-standard setup components based on the subfamily level, and custom component based on the individual PC board level. The component allocation heuristic first attempts to assign standard setup components based on workload requirements of a PCB family. Then the semi-standard components are assigned based on workload requirements of a PCB subfamily and incorporated into the workload of the standard and semi-standard setup components for that subfamily, as determined from previous subfamilies. Finally, the custom-setup components are allocated based on the workload of individual PCBs including the workload due to standard and semi-standard setup components that have already been assigned. The framework of the methodology is shown in Figure 1.

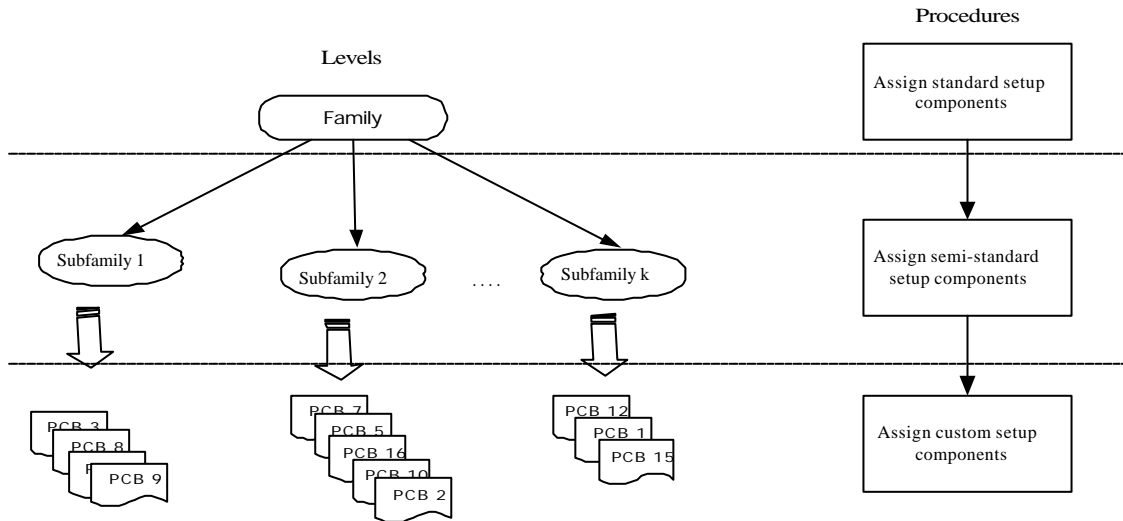


Fig. 1 Component allocation methodology

A brief description of the component allocation algorithm follows:

Step1: Input. Read a sequence of subfamilies in the current family, quantities of components required by individual PCBs, and PCB-to-component incidence matrix.

Step2: Check standard setup components. If there is a set of standard setup components, then (1) read the set of standard setup components for the family, (2) calculate quantities of individual standard setup components for all PCBs in the current family, (3) sort components in decreasing order of these quantities, and (4) initially set workload on machines equal zero; Otherwise, go to step 4.

Step 3: Assign components. If the number of feeder slots already assigned to a machine equal feeder capacity of this machine, then eliminate this machine from consideration. Assign the first component of the sequence to the machine that has the smallest workload. If there is a tie, assign the component to the machine with smallest

index. Then update workload in that machine, remove the assigned component from the sequence, and calculate the number of used feeder slots. Repeat this step until all components in the sequence are distributed.

Step 4 : Check semi-standard setup components. If there is a set of semi-standard setup components at the current subfamily, then (1) read the set of semi-standard setup components that need to be loaded before producing the current subfamily, (2) find the quantities of these semi-standard components needed for one setup, (3) sort these components in order of decreasing quantities, and (4) calculate machines' workload due to standard and semi-standard setup components that already assigned to the current subfamily; otherwise, go to step 6.

Step 5: Assign components. (see step 3)

Step 6: Check custom setup components. If there is a set of custom setups for the current subfamily, then go to step 7; otherwise, go to step 11.

Step 7: Check the number of PCBs. If there is more than one PCB type in this subfamily, then sort the PCB in the order of all components required by each PCB (the biggest PCB lot will have higher priority).

Step 8: Check unassigned custom setup components. Start with the first PCB in the sequence. If there is unassigned custom setup component for this PCB, then (1) sort unassigned custom components (needed to be installed for producing the current PCB) in decreasing quantities based on their usages for this subfamily, and (2) calculate machines' workload due to standard, semi-standard and custom setup components that already assigned to the current subfamily; otherwise go to step 10.

Step 9: Assign components. (see step 3)

Step 10: Check the last PCB. If the current PCB is the last PCB in the subfamily, then go to step 11; otherwise, remove the current PCB from the sequence and consider the next PCB and then go to step 8.

Step11: Check the last subfamily. If this subfamily is the last subfamily, then start new family and go to step 1; otherwise, consider the next subfamily and go to step 4.

2.4 Example Illustration

An example data set with seven PCBs and fifteen components presented in Table 1 is used to explain the allocation algorithm. This table represents the quantity of each component required for each PC board types. The PCB-to-component matrix presented in Table 2 (a), is used to determine the three types of setup components. The setup component types and other production information are summarized in Tables 2 (b) and Table 2(c). There are two identical machines, called M1 and M2, each with five feeder slots. The feeder setup time and placement time for a component are 1.6 and 0.01 units, respectively, for both machines.

Table 1 Quantity of components for each PCB

PCB	Components														
	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15
P1	15	10	0	0	10	0	10	5	20	5	0	5	5	0	0
P2	0	0	5	0	10	20	0	5	0	0	5	5	15	0	0
P3	0	0	0	5	0	0	0	10	0	0	5	0	5	0	0
P4	0	5	0	10	0	0	0	10	0	0	0	0	20	5	0
P5	0	5	5	5	0	15	0	10	0	10	0	0	0	0	0
P6	0	0	10	0	10	0	0	0	0	15	0	0	0	0	0
P7	0	10	10	0	10	5	0	0	0	5	0	10	0	5	10

Table 2 (a) PCB-to-component matrix after executing the KTNS procedure

Sequenced Subfamilies	PCB	Components														
		c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15
1	P7	0	1	1	0	1	1	0	0	0	1	0	1	0	1	1
2	P2	0	1	1	1	1	1	0	1	0	1	1	1	1	0	0
	P5	0	1	1	1	1	1	0	1	0	1	1	1	1	0	0
3	P3	0	1	1	1	1	0	0	1	0	1	1	1	1	1	0
	P4	0	1	1	1	1	0	0	1	0	1	1	1	1	1	0
	P6	0	1	1	1	1	0	0	1	0	1	1	1	1	1	0
4	P1	1	1	0	1	1	0	1	1	1	1	0	1	1	0	0

1 : component associated with column will be in a feeder slot for producing the PCB associated with row

Table 2 (b) Production information

Sequenced Subfamilies	Sub-families	PCB	Lot sizes	Feeder setup		
				Standard	Semi-standard	Custom
1	s4	P 7	10	c2, c5,c10,c12	c3, c6	c14,c15
2	s3	P 2	10	c2, c5,c10,c12	c3, c4, c6, c8, c13, c11	-
		P 5	10			
3	s2	P 4	10	c2, c5,c10,c12	c3, c4, c8, c11, c13	c14
		P 6	10			
		P 3	10			
4	s1	P 1	10	c2, c5,c10,c12	c4, c8, c13	c1, c7, c9

Table 2 (c) the number of required feeder slots

Components	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15
Number of feeder slots	1	1	1	1	1	1	1	2	1	2	1	2	2	1	1

Iteration 1: Read the sequence of subfamilies and the associated production data from Table 2(b), and the quantities of components required by each PCB as shown in Table 1. There are four standard setup components, c2, c5, c10 and c12.

Iteration 2: Calculate the usages of the individual components for all PCBs in the family (see Table 3(a)); then sort them in order of decreasing usage. The component sequence based on decreasing usages is c5, c10, c2 and c12.

Table 3 (a) Calculation of the standard component usage for the first subfamily

Sequenced Subfamilies	Components			
	c2	c5	c10	c12
1	100	100	50	100
2	50	100	100	50
3	50	100	150	0
4	100	100	50	50
Total	300	400	350	200

Iteration 3: Allocate these components to machines on the smallest workload basis. Initially, no component is assigned to any machine; therefore, the workload on both machines is zero. The first component in the order (c5) is allocated to machine M1 followed by the second component (c10) to machine M2. The workloads of the two machines due to assigning components c5 and c10 are $(1 \times 1.6 + 400 \times 0.010) = 5.60$ and $(1 \times 1.6 + 350 \times 0.01) = 5.10$, respectively (see Table 3 (b)). Since the workload on machine M2 after these assignments is smaller, the next component, c2, is assigned to this machine. The procedure is repeated until all components are assigned. In this case, components c5 and c12 are assigned to machine M1 and components c2 and c10 are assigned to machine M2 with 3 used feeders on both machines.

Table 3 (b) Allocation of the standard setup component to machines

Components	Number of components	Workload	TW1	TW2
			0	0

c5	400	5.6	5.6	0
c10	350	5.1	5.6	5.1
c2	300	4.6	5.6	9.7
c12	200	3.6	9.2	9.7

Iteration 4: Since components c3 and c6 are the semi-standard for the first subfamily in the sequence, these components have to be sorted based on their usage for subfamilies they serve. These usages are shown in Table 3 (c). This sequence is c6 followed by c3.

Table 3 (c) Calculation of the semi-standard usage

Sequenced Subfamilies	Components	
	c3	c6
1	100	50
2	100	350
3	100	0
4	0	0
Total	300	400

Iteration 5: Semi-standard components c6 and c3 are now allocated to machines by considering workload of each machine. The current subfamily is the first subfamily to be produced; thus, no semi-standard component has previously been installed. The updated workload on each machine before assigning the semi-standard setup components will consider only standard setup components. Three feeder setups are required for standard setup components c5 and c12 on machine M1 and for c2 and c10 on machine M2. For sequenced subfamily 1, the total usage of components c6 and c12 is $(100+100) = 200$ and the total usage of components c2 and c10 is $(100+50) = 150$ (Table 3 (a)). The total workload of the two machines due to standard setup components are 3.6 and 3.1, respectively. Components c3 and c6 are assigned to machines M1 and M2, respectively. At this point, the components on machines M1 are {c5, c12, c3} and on machine M2 are {c2, c10, c6} with four feeder setups on both.

Iteration 6: PCB 7 is the only PC board type in the first sequenced subfamily and this PCB requires two custom setup components, c14 and c15. Based on their usages, component c15 will be assigned to a machine before component c14. The workload due to semi-standard and standard components on machine M1 and M2 are $(3*1.6+300*.01) = 7.8$ and $(3*1.6+200*.01) = 6.8$, respectively. The components c14 and c15 are assigned to machines M1 and M2. Thus, the sets of components on machines M1 and M2 are {c5, c12, c3, c14} and {c2, c10, c6, c15}, respectively.

Iteration 7: The procedure is repeated to assign semi-standard and custom setup components for the next sequenced subfamilies. The result of component allocation on machines is summarized in Table 4.

Table 4 Result of the component allocation problem

Sequenced Subfamilies	Components														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	2	1	0	1	2	0	0	0	2	0	1	0	1	2
2	0	2	1	1	1	2	0	1	0	2	2	1	2	0	0
3	0	2	1	1	1	0	0	1	0	2	2	1	2	2	0
4	2	2	0	1	1	0	2	1	1	2	0	1	2	0	0

Note : “1” and “2” represent the column component required by associated row sequenced subfamily will be assigned to machines M1 and M2, respectively. “0” represents No column component required by associated row sequenced subfamily is assigned.

3. Evaluation

In order to examine performance of the algorithm, literature test problem was employed to test the effectiveness and industry test problem was also used to study the applicability with large-scale problems. In addition, effects of some key parameters were analyzed.

3.1 Evaluation Using Literature Test Problem

To investigate the quality of the heuristic solution by comparing its solution to a known optimal, part of the data set from [13] is taken with 7 PCB types and 15 component types. Optimal solutions from enumerative method and the heuristic solutions with different scenarios are presented in Table 5. The maximum difference between the optimal and heuristic solutions is 10.76% and the minimum difference is 4.51 %.

Table 5 Comparison of Results with eight cases of various production scenarios

Case	Production Scenarios	Placement time	Optimal Solution	Heuristic Solution	Difference (%)
1	$T_{\text{setup}} = T_{\text{TPT}}$	$PT_{m1} = PT_{m2}$	32.10	34.35	7.01
2	$T_{\text{setup}} = T_{\text{TPT}}$	$PT_{m1} > PT_{m2}$	37.20	45.00	8.45
3	$T_{\text{setup}} = T_{\text{TPT}}$	$PT_{m1} < PT_{m2}$	38.00	41.75	10.01
4	$T_{\text{setup}} < T_{\text{TPT}}$	$PT_{m1} = PT_{m2}$	21.60	24.09	10.76
5	$T_{\text{setup}} < T_{\text{TPT}}$	$PT_{m1} > PT_{m2}$	24.23	26.02	7.39
6	$T_{\text{setup}} < T_{\text{TPT}}$	$PT_{m1} < PT_{m2}$	27.91	30.55	9.46
7	$T_{\text{setup}} > T_{\text{TPT}}$	$PT_{m1} = PT_{m2}$	40.20	42.35	5.35
8	$T_{\text{setup}} > T_{\text{TPT}}$	$PT_{m1} > PT_{m2}$	43.95	47.88	8.94
9	$T_{\text{setup}} > T_{\text{TPT}}$	$PT_{m1} < PT_{m2}$	46.83	48.94	4.51

Note: (1) T_{setup} : Feeder setup time of a component, (2) T_{TPT} : Average placement time per board, and (3) PT_{m1} and PT_{m2} : placement time for a component on machine 1 and 2 respectively.

3.2 Evaluation Using Industry Data

To demonstrate applicability of the heuristic algorithm to solve real world problems and to examine impacts of some key parameters on the solutions, seven data sets from an electronic company are used here. The number of PCB types ranges from 548 to 843 and the number of component types ranges from 281 to 826. The distributions of component usage and PCB requirement are exponentially distributed. An example of distributions for one of these data sets are shown Figure 2 (a) and 2 (b) with means of 26.29 and 32.66, respectively.

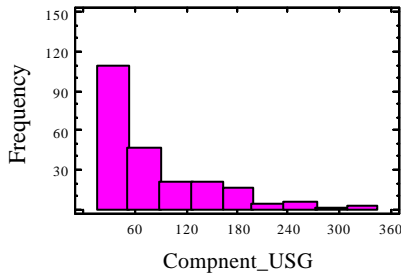


Figure 2 (a) Component usage distribution

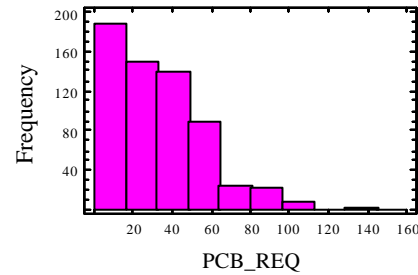


Figure 2 (b) PCB requirement distribution

Note : Component usage (Component_USG) is the total number of PCB types that demand a specific component. The PCB requirement (PCB_REQ) is the total number of component types for a PCB assembly.

The component allocation heuristic attempts to reduce imbalance in both setup and placement time. However, two parameters, threshold and feeder capacity of machines would influence the allocation problem. Threshold, a crucial parameter in the grouping procedure, impacts size of a family [3, 19]. Feeder capacity places limitations on subfamily size in the decomposition procedure [14, 15]. The effects of these parameters on the machines' imbalance are examined with 10 trials, five threshold levels, and two feeder capacity levels. The threshold levels vary between 0.05 and 0.25, in increments of 0.05. The three feeder capacity levels are 150 and 300. From the characteristic of data sets, there are variations of component usage and PCB requirement distributions among data sets; thus, the impact of difference in data sets was also considered.

Statistical analysis is summarized in Table 6. The feeder capacity (CAPACITY) and difference in data sets (DATA_SET) statistically affects total imbalance (IMBALANCE) at a significance level of 0.05, whereas threshold (THRESHOLD) does not influence the total imbalance. The two-way interactions among these factors do not impact IMBALANCE. The heuristic procedure works efficiently; a trial is performed, on average, in 109 seconds on a 800MHz machine. Due to the impacts of DATA_SET and CAPACITY on IMBALANCE, effect of feeder capacity for individual data sets are explored in Table 7. For most of data sets, feeder capacity (CAPACITY) impacts on total imbalance (IMBALANCE). In addition, large feeder capacity tends to increase total imbalance. In large feeder capacity environment there are more components, already assigned on each feeder carriage, effecting assigning a component on a machine than in case of small feeder capacity.

Table 6 ANOVA on IMBALANCE for industry data

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
CAPACITY	1	398911311	398911311	5.65	0.0257
THRESHOLD	4	228499625	57124906.1	0.81	0.5314
DATA_SET	6	2.913E+09	485582465	6.88	0.0002
Capacity*THRESHOLD	4	207231080	5180770.0	0.73	0.5777
CAPACITY*DATA_SET	6	488068822	81344803.7	1.15	0.3634
THRESHOLD*DATA_SET	24	1.582E+09	65916890.8	0.93	0.5656

Table 7 Effect of CAPACITY on IMBALANCE for individual data sets

Data Set	P-value	Significance ($\alpha = 0.05$)
1	<0.0001	Yes
2	0.2875	No
3	0.1097	No
4	0.0055	Yes
5	0.0180	Yes
6	0.0440	Yes
7	0.0406	Yes

4. Conclusion

The heuristic algorithm with three categories of setup components, standard, semi-standard, and custom, is developed to allocate components on machines. Published data sets were then utilized to evaluate the performance of the heuristics. Seven industry data sets were also used to test the applicability of the heuristic and examine the effect of two critical parameters, similarity and feeder capacity, on the total imbalance of production time between machines. The heuristic procedure for the component allocation problem produced good results with the deviations of the heuristic solution from the optimal solution for the tested problem being between 4.5 and 10.8%. Threshold parameter does not have a statistically significant effect on the total imbalance. Size of the feeder capacity has a significant effect on the total imbalance. Large feeder capacity tends to have a higher total imbalance.

References

- [1] D. Ben-Arieh, M. D. Dror, *International Journal of Production Research*, Vol. **28**(7), pp 1317-1327, 1990.
- [2] A. Barnea, D. Sipper, *Computer-integrated Manufacturing Systems*, Vol. **6**(1), pp 18-26, 1993.
- [3] Y. Li, PhD. dissertation, Industrial and Manufacturing Engineering, Oregon State University, OR., USA., 1999.
- [4] R. L. Myers, in *Circuit Assembly*, Vol. 8, pp. 66-68, 1997.
- [5] J. C. Ammons, Carlyle, M., Cranmer L., DePuy G., Ellis K., McGinnis L.F., Tovey C.A., and Xu H., *IE Transactions*, Vol. **29**(1-6), 265-275, 1997.
- [6] G. W. DePuy, PhD dissertation, Industrial and System Engineering, Georgia Institute of Technology, GA, USA, 1995.
- [7] J. Smed, M. Johnsson, M. Puranen, T. Leipala, O. Nevalainen, *Robotics and Computer-integrated Manufacturing*, Vol. **15**(1), pp39-49, 1999.
- [8] Z. Xu, K. Carlson, R. Kurschner, Y. Li, S. Randhawa, *Journal of Electronic Manufacturing*, Vol. **8**(3-4), pp 225-234, 1999.
- [9] M. Gronalt, R. Zeller, *International Journal of Production Research*, Vol. **38**(2), pp 409-427, 2000.
- [10] R. Askin, M. Dror, A. J. Vakharia, *Naval Research Logistics*, Vol. **41**(4), pp 587-608, 1994.

- [11] T. Hayrinen, M. Johnsson, T. Johtla, J. Smed, O. Nevalainen, *Production Planning & Control*, Vol. **11**(5), pp 497-510, 2000.
- [12] T. F. Carmon, O. Z. Maimon, E. M. Dar-EL, *International Production Research*, Vol**27**(8), pp 1795-1810 1989.
- [13] S. Hashiba, T. Chang, *Computers and Industrial Engineering*, Vol. **21**(1-4), pp 453-457, 1991.
- [14] O. Maimon, A. Shtub, *International Journal of Production Research*, Vol. **29**(7), pp 1379-1390, 1991.
- [15] L. F. McGinnis, et al., *IIE transactions*, Vol.**24**(4), pp 18-46, 1992.
- [16] M. Garetti, A. Pozzetti, R. Tavecchio, *production planning & control*, Vol. **7**(2), pp 197-204, 1995.
- [17] K. Rajkumar, T. T. Narendran, *Production Planning & Control*, Vol. **9**(5), pp 465-476, 1998.
- [18] O. Z. Maimon, D. Braha, *International Journal of Production Research*, Vol.**36**(3), pp 761-784, 1998.
- [19] D. R. Sule, *International Journal of Production Research*, Vol.**30**(5), pp 1191-1208, 1992.
- [20] A. Shtub, and Maimon O., *International Journal of Production Research*, Vol. **30**(5), pp 973-983, 1992.
- [21] C. S. Tang, E. V. Denardo, *Operations Research*, Vol.**36**(5), pp 767-777 1988.