An Aggregated Optimization Model for Multi-Head SMD Placements

Ashayeri, J.; Ma, N.; Sotirov, R.

Publication date:
2010

CentER

Discussion Paper

No. 2010–46

AN AGGREGATED OPTIMIZATION MODEL FOR MULTI-HEAD SMD PLACEMENTS

By J. Ashayeri, N. Ma, R. Sotirov

April 2010

ISSN 0924-7815

TILBURG ◆ UNIVERSITY

An Aggregated Optimization Model for Multi-head SMD Placements

J. Ashayeri* N. Ma† R. Sotirov‡

May 4, 2010

Abstract

In this article we propose an aggregate optimization approach by formulating the multi-head SMD placement optimization problem into a mixed integer program (MIP) with the variables based on batches of components. This MIP is tractable and effective in balancing workload among placement heads, minimizing the number of nozzle exchanges, and improving handling class. The handling class which specifies the traveling speed of the robot arm, to the best of our knowledge, has been for the first time incorporated in an optimization model. While the MIP produces an optimal planning for batches of components, a new sequencing heuristics is developed in order to determine the final sequence of component placements based on the outputs of the MIP. This two-stage approach guarantees a good feasible solution to the multi-head SMD placement optimization problem. The computational performance is examined using real industrial data.

Keywords: Multi-head surface mounting device, Component placement, Variable placement speed

JEL-code: C6

1 Introduction

The multi-head surface mounting device (SMD) (see Ayob and Kendall 2008) is one of the most popular auto-assembly machines due to its relative high speed in mounting components on printed circuit board (PCB) and low price. The optimization problem for improving the throughput of its operations, however, is shown to be highly complex. McGinnis et al. (1992) summarize optimization problems for a SMD as arrangement of component feeders and sequencing of placement operations. To be specific, the major optimization problems of a multi-head SMD production planning consist of feeder arrangement, component and nozzle assignment to each placement head, as well as sequence of component placements on PCB. In addition to these problems, in this paper,

*Department of Econometrics and OR, Tilburg University, The Netherlands. Email: j.ashayeri@uvt.nl
†Department of Econometrics and OR, University, The Netherlands. Email: n.ma@uvt.nl
‡Department of Econometrics and OR, University, The Netherlands. Email: r.sotirov@uvt.nl

we are also interested in improving the traveling speed of the robot arm, which is a function of component delivery nozzles mounted on the robot placement heads. We specify this as a handling class (HC). Namely, some nozzles are better in handling certain component type and allow higher traveling speed of robot arm. Therefore, HC is implicitly determined as a result of component and nozzle assignment to the placement heads. Hence extra attention in component and nozzle assignment is needed in order to guarantee optimal HC. The HC problem or the traveling speed problem addressed in this paper, to the best of our knowledge, has been for the first time incorporated in an optimization model.

Among the major problems of a SMD production planning, feeder arrangement is one of the crucial problems impacting PCB assembly throughput time. Lee et al. (2000) develop a model solved by dynamic programming for determining the feeder assignment of a multi-head SMD and provide a method for reducing the computation time. Ayob and Kendall (2005) focus on improving the feeder setup of a sequential pick and place machine in order to minimize the robot assembly time, the feeder movements and PCB table movement. Li et al. (2008) studied an application of genetic algorithm for obtaining a feeder assignment of a turret-type SMD. Duman and Or (2007) search among specific algorithms reflecting implementation of taboo search, simulated annealing and genetic algorithm-type metaheuristics in order to identify the well performing heuristic procedures for solving the quadratic assignment problem of feeder assignment. Duman and Or (2007) conclude that the performance of a heuristic highly depends on the problem specifications.

The feeder arrangement for our assembly problem is formulated before conducting of the proposed research in this paper. Therefore, we assume that the planning solution to the feeder assignment is known and focus on the remaining major problems of a multi-head SMD production planning. We refer to the remaining major problems, which consist of component and nozzle assignment to each placement head, improving HC, and sequence of component placements on PCB, as our multi-head SMD placement optimization problem. Very few literature has the same problem setting as ours because of the diversity of the machine type and the range of complexity problems involved. Burke et al. (1999, 2001a,b) formulate a generalized traveling salesman problem model based on hypertours for a SMD which has the similar feature as ours. Their formulation includes the considerations of component type assignment to feeder slots, tool assignment to placement locations, and component placement sequence. A constructive and local search heuristics is provided in order to reduce computation time and determine locally optimal solutions. Lee et al. (2000) develop a hierarchical approach considering following three subproblems: construction of feeder reel-groups, assignment of those feeder reel-groups, and sequencing of pick-and-place movements, each of which is solved by a heuristics. The proposed method can be applied to SMD with any number of heads. Knuutila et al. (2007) proposed a greedy heuristic under the multi-head SMD environment for nozzle selection with the aim of minimizing the number of pickups when the sequence of component placements is given. This heuristic produces optimal solution under restricted assumptions. One observation from the literature review is that almost every kind of mathematical formulations related to the multi-head SMD placement optimization problem turn to be a large scale problem that cannot be solved in a reasonable time frame. Therefore, methods like traveling salesman problem (TSP) heuristic (see Lee et al. 2000, Zeng et al. 2004), local search (see Ayob and Kendall 2003, Burke et al. 1999, 2001a,b), and genetic algorithms (see Hardas et al. 2008, Li et al. 2008, Sun et al. 2005) are frequently applied.

As an effort in pursuing high quality solution, we present a way of deriving a tractable mathematical model for solving the multi-head SMD placement optimization problem. In this paper, we develop a multi-objective MIP model for the multi-head SMD placement optimization problem based on batches of components along with a heuristic placing algorithm. The idea is to determine the optimal sequence of batches of components to the placement heads in the first stage by solving the MIP, and then to determine the sequence of components with a heuristic method in the second stage. These two steps together assure a feasible solution produced in reasonable time.

The rest of the paper is organized as follows. In Section 2, we describe the main features of an auto-assembly process of a multi-head SMD. The MIP model is presented in Section 3, and the heuristic method for determining the final sequence of components is discussed in Section 4. In Section 5 we present the numerical results for the proposed approach with 15 real-life data sets.

2 SMD Auto Assembly Problem

In this paper we are interested in the multi-head SMD of the type AX2.01, see Figure 1, which is developed by Assembléon, formerly known as Philips Electronic Technology. It is a high accurate mounting device which is specialized for placing large number of components on a PCB. It is equipped with a fix PCB table, one feeder bank close to a corner of the PCB table, a single 4-head robot arm, one automatic nozzle changer (ANC) and two extra cameras for alignment. In each pick-and-place cycle, the robot arm moves from feeder banks first to the cameras and then to the PCB table where the mounting operation is taken place. The alignment at the cameras is required for providing high accuracy of the mounting operation, and components are first scanned and rotated if it is necessary for adjusting the positions in order to have components pointed at the planned directions. The cameras can align at most two components at a time. After leaving the PCB table, the robot arm then first visits ANC for exchanging nozzles before going to the feeder banks if there is one or more components in the next pick-and-place cycle requiring a nozzle different than those that are currently in use. The nozzle exchanges are normally very time consuming.

Without loss of generality, following assumptions are made:

1. We assume that each PCB of a certain type is processed one after another by the SMD.
2. The SMD is equipped with a fixed PCB table, one fixed feeder bank placed at low-left corner of the PCB table, an ANC, a robot arm, and a pair of cameras.

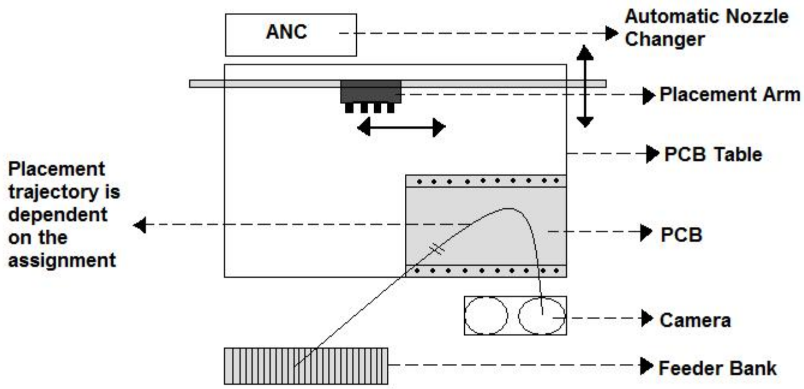

Figure 1: Layout of AX2.01

3. The robot arm has four placement heads and can carry at most four components in one pick-and-place cycle. Note that it is also possible to carry less than four components in one pick-and-place cycle.

4. The pair of cameras take a fixed time for scanning all components carried in one pick-and-place cycle.

5. The robot arm travels from camera to PCB for placing components, then travels to ANC first if nozzle-change is necessary, and then goes to the feeder bank for picking up components in each pick-and-place cycle.

6. The time needed for traveling in between the PCB table, the feeder bank, and camera is assumed to be fixed. Note that the traveling time between the PCB table and the feeder bank is assumed to be identical no matter wether ANC is visited by the robot arm in a pick-and-place cycle or not.

7. Powered by two separate motors, the robot arm travels simultaneously in the horizontal and vertical directions. Note that the separate motors may generate different traveling speeds in the horizontal and vertical directions. In this setting, the traveling time between two points on the PCB table is considered as the maximum of the horizontal and vertical traveling times. We refer to this type of movement as a Chebyshev traveling movement. (According to Abello et al. 2002 the Chebyshev distance is a metric defined on a vector space where the distance between two vectors is the greatest of their differences along any coordinate dimension.)

8. The HC specifies for each component the preferred delivery nozzles and the corresponding traveling speeds for different component-nozzle matchs. There are possibly more than one nozzles that can pick up a certain component.

9. The arm traveling speed is defined by the highest HC among four placement heads. We assume the lower the HC the higher the traveling speed, and the higher the HC the slower the traveling speed.

10. The time for picking up components at the feeder bank is assumed to be identical in every pick-and-place cycle.

11. Nozzles of the same type can be assigned to the placement heads simultaneously.

12. Every placement head is capable of visiting all places on the PCB table.

The main purpose of this research is to minimize the total processing time for mounting a PCB, which includes following four objectives: minimizing the number of nozzle exchanges, balancing workload among four placement heads, maximizing the traveling speed, and minimizing the traveling distance. The hierarchical procedure we propose splits the previously mentioned four objectives into two stages. In the first stage, we formulate a multi-objective MIP problem that includes optimizing the first three above mentioned objectives. In the second stage, we implement a heuristic that is based on the results obtained from the first stage, in order to determine the final sequencing by minimizing the traveling distance.

The main reasons for this partitioning are:

- The first three objectives are highly correlated with each other. Component assignment limits the possible nozzle selection, and the nozzle selection determines the HC and hence the traveling speed.

- The production processing time can be reduced most significantly by reducing the number of nozzle exchanges and the number of pick-and-place cycles, and increasing the average traveling speed. The traveling distance minimization, however, offers the least improvement on reducing the processing time.

- The exclusion of minimizing travel distance from first stage allows MIP formulation based on batches of components instead of single component, which results with reduced complexity.

- The MIP formulation based on batches of components defines the characteristics of the batches assigned to placement heads, such as the batch size, the component type, and the order of placements, but not the allocation of each individual component. Hence, by determining the component-wise placement order and allowing components exchange between batches of the same component type, the optimization in the second stage can still greatly explore the opportunity of refinement.

We believe that this hierarchical procedure is the best optimization approach in terms of reducing the complexity while maintaining optimization as much as possible. We present the inputs and the desired outputs of the hierarchical procedure in the sequel.

The inputs to the hierarchical procedure include:

- Component classification: A table with information on different component types.
- Handling Class: A matrix which specifies the HC for each pair of component type-nozzle match.
- Component location: The x-y coordinates of all components on a PCB.

As a result of the hierarchical procedure, the following information is obtained and a combination of these information is referred to as "chargelist".

- The components assignment to each of the placement heads.
- The placement sequences of the components.
- The nozzle selection for handling each of these components.

3 First Stage: The MIP Model

In this section, we derive the MIP model that solves the first stage of the problem. Variables in our MIP model are based on batches of components. By aggregating variables to the batches, we obtain a model with reduced number of assignment variables. This approach provides a unique alternative paradigm for typical assignment problem in electronic/semi-conductor industries. A batch is defined as a set of identical components that needs to be placed on a PCB by a certain placement head. The total number of identical components can be divided into few batches, if it is justified by the optimization.

Below parameters are used in the model formulation:

I	the number of component types
J	the number of nozzle types
$k \in \{1, 2, 3, 4\}$	the set of placement heads
L	the maximum number of batch levels $L \leq I + 1$
comp_i	the number of identical components of type i
M	a given large number (that is larger than $\max\limits_{i \in I}\{\text{comp}_i\}$)
H_{ij}	the HC when component of type i is handled by nozzle type j

Variables in the model are:

X_{ijk}	the number of components of type i that are placed by nozzle type j on placement head k
N_k	the total number of nozzle exchanges on placement head k
H_l	the worst HC of all batches on level l
WL	the largest workload of four placement heads

$$Z_{ijlk} = \begin{cases} 1, & \text{if batch } X_{ijk} \text{ is placed on level } l \\ 0, & \text{otherwise,} \end{cases}$$

$$D_{lk} = \begin{cases} 1, & \text{if there is a change of nozzle in the level } l+1 \text{ on placement head } k \\ 0.5, & \text{if there are no batches placed on levels higher than } l \\ 0, & \text{otherwise.} \end{cases}$$

Our MIP model is formulated in the sequel:

$$\text{Minimize} \qquad a \cdot WL + b \cdot \sum_{k=1}^{4} N_k + c \cdot \sum_{l=1}^{L} H_l \tag{1}$$

Subject to:

$$\sum_{j=1}^{J} \sum_{k=1}^{4} X_{ijk} = \mathrm{comp}_i \qquad \forall i \tag{2}$$

$$\sum_{j=1}^{J} \sum_{i=1}^{I} X_{ijk} \leq WL \qquad \forall k \tag{3}$$

$$X_{ijk} \leq \mathrm{M} \cdot \sum_{l=1}^{L} Z_{ijlk} \qquad \forall i, j, k \tag{4}$$

$$\sum_{l=1}^{L} Z_{ijlk} \leq 1 \qquad \forall i, j, k \tag{5}$$

$$\sum_{l=1}^{L} Z_{ijlk} \leq X_{ijk} \qquad \forall i, j, k \tag{6}$$

$$\sum_{j=1}^{J} \sum_{i=1}^{I} Z_{ijlk} \geq \sum_{j=1}^{J} \sum_{i=1}^{I} Z_{ij(l+1)k} \qquad \forall k, l \tag{7}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{J} Z_{ijlk} \leq 1 \qquad \forall l, k \tag{8}$$

$$D_{lk} = \frac{1}{2} \cdot \sum_{j=1}^{J} \left| \sum_{i=1}^{I} Z_{ijlk} - \sum_{i=1}^{I} Z_{ij(l+1)k} \right| \qquad \forall k, l \tag{9}$$

$$N_k = \sum_{l=1}^{L} D_{lk} - 0.5 \qquad \forall k \tag{10}$$

$$H_l \geq \sum_{i=1}^{I} \sum_{j=1}^{J} H_{ij} \cdot Z_{ijlk} \qquad \forall l, k \tag{11}$$

$$
\begin{aligned}
Z_{ijlk} \in \{0,1\},\ D_{lk} \in [0,1],& \\
X_{ijk} \in \mathbf{Z}^+,\ N_k \in \mathbf{Z}^+,\ H_l \in \mathbf{R}^+.& \qquad (12)
\end{aligned}
$$

Where a, b, c are real numbers, whose values are determined as described in Section 5.

Constraint (2) guarantees that the sum of the different batch sizes of component type i is equal to the predetermined size of component type i. Constraint (3) calculates the largest workload among placement heads. Constraint (4) ensures that batch X_{ijk} always has a place when its size is greater than zero. Constraint (5) guarantees that batch X_{ijk} can be assigned at most to one place. Constraint (6) ensures that there are no assigned locations for batches of size zero. Constraint (7) ensures that the batches have to be located at lower level l before being located to the higher level $l+1$. Constraint (8) ensures that each place can be used to allocate at most one batch. The introduction of constraint (9) is intended to count for each level of batches whether there is a nozzle exchange. D_{lk} is equal to one when there is an exchange in the next level, and is equal to zero if there is no exchange. Since, based on our formulation, every nozzle exchange will be counted once for each of the nozzle types on different level, the actual counted number is doubled. That is why we multiply 0.5 to the counted value. However, this implementation results in a extra output value of 0.5 for D_{lk} when Z_{ijlk} for some i and j is the last batch assigned to placement head k, i.e., when there are no more batches assigned to higher levels than l. This constraint can be further formulated as below:

$$
\begin{aligned}
\sum_{i=1}^{I} Z_{ijlk} - \sum_{i=1}^{I} Z_{ij(l+1)k} = D^+_{ljk} - D^-_{ljk} & \qquad \forall l, j, k \\
D_{lk} = \frac{1}{2} \cdot \sum_{j=1}^{J} (D^+_{ljk} + D^-_{ljk}) & \qquad \forall k, l \\
D^+_{ljk}, D^-_{ljk} \geq 0 & \qquad \forall l, j, k.
\end{aligned}
$$

Constraint (10) counts the total number of nozzle exchanges on one placement head. Under the assumption of repetitive production, the planned placement order of batches is processed reversely between sequential PCBs, during which the last batch level of the on-processing PCB are processed as the first batch level of the next PCB. Hence, there is no nozzle exchange after processing the last-level batches, and the counted 0.5 times of nozzle exchange at the last batch level can be removed from calculation. Constraint (11) determines the worst HC for each batch level.

From the solution of the optimization problem (1) - (12) we obtain the batch size, the batch location, the nozzle assignment for each batch, as well as the workload on each placement head, the number of nozzle exchanges, and the HC for each pick-and-place cycle. The complexity of this model with respect to the number of variables and constraints is presented below. The complexity depends on the number of component types I, the number of nozzle types J, as well as the maximum number of levels for batches L.

$$\text{Number of Variables} = \begin{cases} I \cdot J \cdot 4 & \text{for} \quad X_{ijk} \\ I \cdot J \cdot 4 \cdot L & \text{for} \quad Z_{ijlk} \\ L \cdot 4 & \text{for} \quad D_{lk} \\ 4 & \text{for} \quad N_k \\ L & \text{for} \quad H_l \\ 1 & \text{for} \quad WL \end{cases} = (4L+4) \cdot I \cdot J + 5 \cdot L + 5$$

$$\text{Number of Constraints} = \begin{cases} I & \text{for (2)} \\ 4 & \text{for (3)} \\ I \cdot J \cdot 4 & \text{for (4)} \\ I \cdot J \cdot 4 & \text{for (5)} \\ I \cdot J \cdot 4 & \text{for (6)} \\ 4 \cdot L & \text{for (7)} \\ 4 \cdot L & \text{for (8)} \\ 4 \cdot L & \text{for (9)} \\ 4 & \text{for (10)} \\ 4 \cdot L & \text{for (11)} \end{cases} = 12 \cdot I \cdot J + 16 \cdot L + I + 8$$

Note that the complexity of the model is greatly influenced by the number of component types rather than the number of components. More precisely, the complexity of MIP model (1) - (12) w.r.t. the number of variables and constraints is $O(I^2 J)$, where I is the number of component types.

When a model for the multi-head SMD placement optimization problem is formulated based on single components, then the complexity of the corresponding single component based model is $O(n^2 J)$, where n is the number of components. We have here in mind a reformulation of problem (1) - (12) where the size of each batch is one, i.e. a batch is a component. Note that for $n \gg I$, the single component based model becomes intractable while our proposed model remains tractable. Following the same argument, the complexity of the aggregated model is never larger than the single based one, since the number of component types is at least the number of components.

4 Second Stage: The Heuristic Method

From the solution of MIP model, we obtain the component type of each batch, the batch sizes, and the order of the batches assigned to each placement head. These information can be presented in a bar chart as Figure 2. Data presented in Figure 2 is the solution of MIP model of the Case 1 from Table 2. The different bars represent the workload on placement heads. The different shaded areas on the bars represent different batches. The hight of each shaded area represents the batch size. The order of batches on a bar indicates the placement order on a certain placement head.

In the second stage, a heuristic is used to determine a placement sequence based on the previously mentioned batch information. The heuristic determines which components need to be grouped into one pick-and-place cycle, the order of pick-and-place

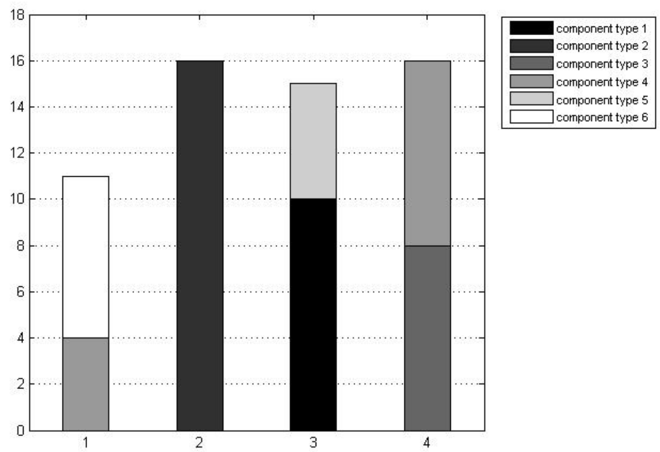

Figure 2: Output of MIP

cycles, and the placement sequence of components in each pick-and-place cycle. This problem is a special case of the traveling salesman problem which is complicated by the Chebyshev traveling feature of the robot arm and the classification of the component type. The traveling problem of Chebyshev feature is well known as the Chebyshev Traveling Salesman Problem (CTSP). As mentioned in Bozer et al. (1990), many heuristic procedures based on geometric concepts have been developed for the CTSP. However, none of the heuristic procedures mentioned in Bozer et al. (1990) deal with the CTSP combined with the classification of vertices needed to be visited, which in our case refers to the classification of the components. Note that grouping components into one pick-and-place cycle and determining the order of pick-and-place cycles is equivalent to the determination of the specific components composing the different batches on the placement heads (shaded areas on each bar in Figure 2) and the placement order of these components in each batch. Instead of formulating another MIP or adapting an existing heuristic to determine the optimal solution, a greedy heuristic named as *level placing algorithm* is developed in order to provide a good feasible solution for our problem.

Our *level placing algorithm* consists of a selection process and an optimization process. The selection process is for grouping components into each pick-and-place cycle such that the selected components can fill in the batches on the placement heads level by level with correct component types. In the selection process, our algorithm selects components from each component type based on the smallest possible y-axis scheme, i.e., the components of the lowest y-coordinate among valid components of different component types are selected first. The optimization procedure determines the best placement sequence of these selected components with respect to the minimum Chebyshev traveling distance. We present our *level placing algorithm* for the final sequencing in Figure 3.

Notations used in the algorithm:

- C_{n_i} stands for the n_i-th component which is of component type i.
- $Y_k^{i_k}$ stands for the number of components of type i_k which are going to be assigned to placement head k in the following pick-and-place cycles.

Combining the results of MIP model and this heuristic method, we complete the placement optimization problem of multi-head surface mounting device. The numerical results are presented in the following section.

5 Numerical results

The MIP model is solved by CPLEX using AIMMS interface, running on a PC with PENTIUM 4, 2.4GHz and 1.5GB of memory. Our heuristic algorithm is implemented by MATLAB 2007b on the same machine.

The a, b, c coefficients of MIP model are determined as $1, 6, 1$ respectively based on the following management judgment: 1 additional minute for one extra pick-and-place cycle, 6 additional minutes for one extra nozzle exchange, and 1 additional minutes for one HC increase. These values are also verified by the factorial design that is described as follows. A factorial design of 27 scenarios with 3 levels for each of the coefficients is conducted based on the data of Case 1 of Table 2. The performance of MIP model is tested for these scenarios and the outputs of the MIP models of these scenarios can be classified into three different categories. We present these three categories in Table 1 in terms of their outputs of MIP models: workload, nozzle exchange, and HC. The last column of Table 1 indicates the number of scenarios which are classified into the same category. More than half of the scenarios fall into the first category. We may note that the MIP model with the proposed combination of coefficients of objective terms $1, 6, 1$ provides the same outputs as in the first category. Hence, the management judgment is producing moderately robust results. The closeness of the outputs of MIP models of these categories also indicates a sufficient efficiency of MIP model in terms of high tolerance of a, b, c estimation errors. However, it is critical that the effects of coefficients' value choice be verified as clearly as possible before the final solution.

Category	Workload	Nozzle exchange	HC	$\sharp$ of scenarios with same outputs
1	16	0	1 and 5	14
2	15	1	1 and 5	9
3	15	0	1 and 8	4

Table 1: Factorial Design Results

Algorithm

Input:

the number of components on PCB: N

the number of component types: I

the number of components of type i: $N_i, \forall i \in \{1, 2, 3, \ldots, I\}$

component C_{n_i} with coordinates $(x_{n_i}, y_{n_i}), \forall n_i \in \{1, 2, \ldots, N_i\}$

output from MIP: X_{ijk}, $Z_{ijlk}, \forall i, j, l, k$

the number of batches on each placement head k, $L_k = \sum_i \sum_j \sum_l Z_{ijlk}, \forall k = 1, 2, 3, 4$

available components of type i, $A_i = \{C_{n_i} | \, n_i \text{ is of type } i\}, \forall i \in \{1, 2, 3, \ldots, I\}$

Sort C_{n_i} in A_i nondecreasingly w.r.t. $y_{n_i}, \forall i$ **End Sort**

$l_k \leftarrow 1, \forall k = \{1, 2, 3, 4\}$

$Y_k^{i_k} \leftarrow \sum_i \sum_j Z_{ijl_k k} X_{ijk}, \forall k = \{1, 2, 3, 4\}$

Total Chebyshev Travel Distance: $TCTD \leftarrow 0$

While $Y_k^{i_k}$ is larger than zero for at least one $k \in \{1, 2, 3, 4\}$

  [P1] choose and remove the first components from corresponding available
      component sets A_{i_k} and set $Y_k^{i_k} \leftarrow Y_k^{i_k} - 1, \forall k \in \{k | \, Y_k^{i_k} > 0\}$

  [P2] calculate the Chebyshev travel distance d_s for every possible
      placing sequence s of these chosen components
      (24 alternatives when 4 components are chosen)

  [P3] travel distance $d_{min} = \min\limits_s d_s$ and set the total Chebyshev
      travel distance $TCTD \leftarrow TCTD + d_{min}$

  If $Y_k^{i_k}$ equal to zero for one or more k and $l_k < L_k$

    [P4] $l_k \leftarrow l_k + 1$ and set $Y_k^{i_k} \leftarrow \sum_i \sum_j Z_{ijl_k k} X_{ijk}, \forall k \in \{k | \, Y_k^{i_k} = 0\}$

  End If

End While

Output

$TCTD_{optimal} = TCTD$

Figure 3: Level Placing Algorithm

In Table 2, we present the numerical results of 15 real-life data sets based on the above coefficient estimation. Note that the data set of Case 1 is created for the demonstration and analysis purpose. In first three columns of Table 2, characteristics of each of the cases are specified. This includes the number of component types, the number of components, and the lowest and highest possible HC. In "case complexity", the number of variables and the number of constraints are specified for each case. These two values determine the complexity of the corresponding computation. The test results from the first stage and the second stage are listed in "result of first stage" and "result of second stage" respectively, which include the computation time, the final workload, the number of nozzle exchanges, the lowest and highest HC in chargelist, and the final traveling distance. Based on the results, there are few observations we would like to highlight in the rest of this section.

- Our method indeed provides a good solution in terms of balancing the workload, minimizing the number of nozzle exchanges and improving HC in the final chargelist.
    1. In Case 2, 3, 4, 9, and 12, all the components are assigned onto 4 placement heads optimally. The workloads are balanced optimally.
    2. The nozzle exchanges are almost always avoided except in Case 11, 12 and 16. The number of nozzle exchanges is effectively minimized.
    3. The HC are improved clearly in Case 1, 4, 15. The highest HC is reduced from 8 to 5 in Case 1, from 4 to 2.7 in Case 4 and from 8 to 2 in Case 15. Note that we only present and examine the highest HC reduction in Table 2, although the results do not explicitly indicate improvements for the rest of cases, the actual improvements are significant for most cases.
- The computation time of the first stage increases as the number of variables and constraints increase. Note that the number of variables increases along with the number of component types rather than the number of components. When the number of variables are below 300, the computation time is less than one second, like in Case 2, 3, and 4. When the number of variables does not exceed 1000, the computation time is in a range of few seconds. When the number of variables increases to a range between 1000 and 2000, the computation is still manageable, and the computation time may vary from 1 minute to few hours. When the number of variables exceed 2000, the computation time may increase to days and even weeks when there are over 5000 variables.
- The number of components does not influence heavily the computation time of a problem. As for an example when comparing Case 2 and Case 4, although Case 2 has more than four times as many components than Case 4, there is hardly any increase in the computation time of Case 2. Case 11 with 600 components experiences a rather long computation time which is caused by the substantially enlarged feasible region (see constraint (2) of page 7).

	Case Characteristics			Case Complexity		Result of 1st Stage			Result of 2nd Stage
case	♯ comp. type	♯ comp.	L/H HC	♯ var.	♯ of constr.	WL	nozzle exch.	L/H HC	total dist.(m)
1	6	58	1/8	477	338	16	0	1/5	5.5632
2	1	80	2/2	64	85	20	0	2/2	7.6139
3	1	40	2/2	81	105	10	0	2/2	3.7145
4	4	16	2/4	285	224	4	0	2/3	1.6588
5	7	21	2/2	738	451	6	0	2/2	2.0100
6	7	12	2/2	427	295	6	0	2/2	1.0096
7	8	9	2/3	359	260	3	0	2/3	0.9899
8	8	15	2/8	859	488	5	0	2/8	1.1273
9	9	43	2/4	1149	725	11	0	2/4	4.0568
10	11	13	2/4	1341	635	4	0	2/4	1.6928
11	13	600	2/8	1205	581	150	1	2/8	56.9631
12	13	64	2/2	1772	769	16	1	2/2	5.9389
13	14	22	4/8	1435	634	8	0	4/8	2.6632
14	19	27	2/8	2568	943	8	0	2/8	3.5121
15	24	54	2/8	5660	1584	15	0	2/2	7.9704
16	28	80	2/8	5076	1516	24	3	2/8	11.0438

Table 2: Basic Results

Case	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Largest/Smallest Batch Size	10/10	20/20	1/8	1/10	2/12	4/4	1/8	1/2	3/3	1/4	1/2	1/5	1/5	3/114	1/3	3/16

Table 3: Batch Size of Basic Results

- The computation time of the second stage is negligible. Even when the component number increase to 600 in Case 11, the computation time is only about 2 seconds.
- The total traveling distance as an output of the second stage is the sum of Chebyshev distances of sequentially movements of the robot arm. We may see that total traveling distance increases as the number of components increases. The total traveling distance can be rescaled into total traveling time by dividing the sequential Chebyshev distances with the corresponding traveling speeds on those directions.
- We observe that the batch sizes could vary greatly from only one component to almost 114 components (see Table 3), which are not constrained in our formulation.

6 Conclusion

The focus of this paper is to explore the use of aggregate modeling in assigning and sequencing of batches of components to a multi-head SMD.

We propose a hierarchical approach of two stages for solving the PCB component placement problem. The MIP model is based on variables of batches of components rather than individual components. This way of modeling turns to be valuable for developing a tractable mathematical programming model in terms of reducing the computation time and shows a great effectiveness in balancing the workload, minimizing the number of nozzle exchanges, and improving HC. The numerical results indicate that this MIP model is most efficient for PCB types which has large number of components but limited number component types. The computation time needed for *level placing algorithm* providing the final sequencing is negligible.

Based on this hierarchical approach, a near optimal solution is reached in reasonable computation time. The outputs of this approach can be used in industry as an high quality solution of an off-line optimization, which can be further tested and improved by on-line optimization techniques.

References

Abello, J., Pardalos, P.M., Resende, M.G.C., 2002. Handbook of Massive Data Sets. Springer.

Ayob, M., Kendall, G., 2003. Real-time scheduling for multi headed placement machine, in: Proceedings of the 5th IEEE International Symposium on Assembly and Task Planning, ISATP'3, Besançom, pp. 128–133.

Ayob, M., Kendall, G., 2005. A triple objective function with a chebychev dynamic pick-and-place point specification approach to optimise the surface mount pacement machine. European Journal of Operational Research 164, 609–626.

Ayob, M., Kendall, G., 2008. A survey of surface mount device placement machine optimisation: Machine classification. European Journal of Operational Research 186, 893–914.

Bozer, Y., Schorn, E., Sharp, G., 1990. Geometric approaches to solve the chebyshev traveling salesman problem. IIE Transactions 22, 238–254.

Burke, E., Cowling, P., Keuthe, R., 1999. New models and heuristics for component placement in printed circuit board assembly, in: Proceedings of the 1999 IEEE International Conference on Information, Intelligence and Systems (ICIIS99), pp. 133–140.

Burke, E., Cowling, P., Keuthe, R., 2001a. Effective heuristic and metaheuristic approaches to optimize component placement in printed circuit board assembly, in: Proceedings of the First European Workshop on Evolutionary Computtion in Combinatorial Optimization EvoCOP2001, Lecture Notes in Computer Science, Springer, Berlin.

Burke, E., Cowling, P., Keuthe, R., 2001b. The printed circuit board assembly problem: Heuristic approaches for multiheaded placement machinery, in: Proceedings of the International Conference on Artificial Intelligence (ICAI2000), CSREA Press, Las Vegas. pp. 1456–1462.

Duman, E., Or, I., 2007. The quadratic assignment problem in the context of the printed circuit board assembly process. Computers and Operations Research 34, 163–179.

Hardas, C., Doolen, T., Jensen, D., 2008. Development of a genetic algorithm for component placement sequence optimization in printed circuit board assembly. Computers and Industrial Engineering 555, 165–182.

Knuutila, T., Pyöttiälä, S., Nevalainen, O., 2007. Minimizing the number of pickups on a multi-head placement machine. Journal of the Operational Research Society 58, 115–121.

Lee, S., Lee, B., Park, T., 2000. A hierarchical method to improve the productivity of a multi-head surface mounting machines. Intelligent Automation and Soft Computing 6, 91–301.

Li, S., Hu, C., Tian, F., 2008. A hierarchical method to improve the productivity of a multi-head surface mounting machines. Applied Soft Computing 8, 522–529.

McGinnis, L., Ammons, J., Carlyle, M., Cranmer, L., Depuy, G., Ellis, K., Tovey, C., Xu, H., 1992. Automated process planning for printed circuit card assembly. IIE Transactions 24, 18–30.

Sun, D., Lee, T., Kim, K., 2005. Component allocation and feeder arrangement for a dual-gantry multi-head surface mounting placement tool. International Journal of Production Economics 95, 245–264.

Zeng, Y., Ma, D., Jin, Y., Yan, J., 2004. Hierarchical planning for a surface mounting machine placement. Journal of Zhejiang Univiversity - Science A 5, 1449–1455.