

Virtual Prototyping of Electronics Assembly Systems

D. A. Bodner, M. Damrau, P. M. Griffin, L. F. McGinnis and A. McLaughlin
Keck Virtual Factory Lab
School of Industrial & Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205, USA
douglas.bodner@isye.gatech.edu

Abstract

Electronics assembly systems typically are capital intensive and highly automated. At the same time, due to product sophistication and complexity, they are difficult to design, configure and reconfigure, and operate. A significant body of research has investigated problems involved in these areas, with the goals of cycle time minimization and line balancing. Equally important is the ability to experiment with different designs, configurations and control policies off-line, and to test the results of algorithms developed to minimize cycle times and balance assembly lines. This requires development of high-fidelity models of system behavior to serve as virtual prototyping environments. In this paper, we describe our work to date in developing such models for single machines and in using these models in an instructional setting to aid student understanding and learning.

Keywords

Virtual prototyping, simulation, 3D animation, electronics assembly, experimental platform, system performance.

1. Introduction

Electronic products are increasingly widespread in the marketplace. At the same time, they are becoming more sophisticated and customized. Therefore, systems that produce them are under pressure to meet throughput requirements and produce multiple product types. This situation creates a number of complex decision problems that engineers must solve in the design, operation and control of assembly systems. This paper concentrates on the decision problems associated with individual machines that assemble electronic components onto printed circuit boards (PCBs). In particular, it focuses on models that evaluate solutions to these problems in terms of machine cycle time. The goal is to develop the ability to study machines by using a *virtual prototype*, or highly realistic model of machine behavior. While a virtual prototype is not a virtual reality model, it does have a three-dimensional animated representation of machine behavior.

In the class of machines studied here, each machine has three mechanisms that interact with one another to perform attachment, or placement, of components. The *component feeder* houses the components to be assembled. Each component type may occupy one or more slots on the feeder, depending on component size. The *board locator* holds the PCB during assembly. The *placement mechanism* picks components from the feeder and attaches them to the PCB. This mechanism has a set of heads that perform pick and place operations. Each head has a nozzle that grips the components via a vacuum. Different size components require different size nozzles.

In some machines, the component feeder is a carriage that moves to position a component to be picked for placement. Likewise, in some machines the board locator moves to position the board correctly for placement of a component. In other types of machines, these two mechanisms are stationary, and the placement mechanism performs all movement. We will assume the latter type of machine. Figure 1 shows an abstract view of a machine (often called a placement machine).

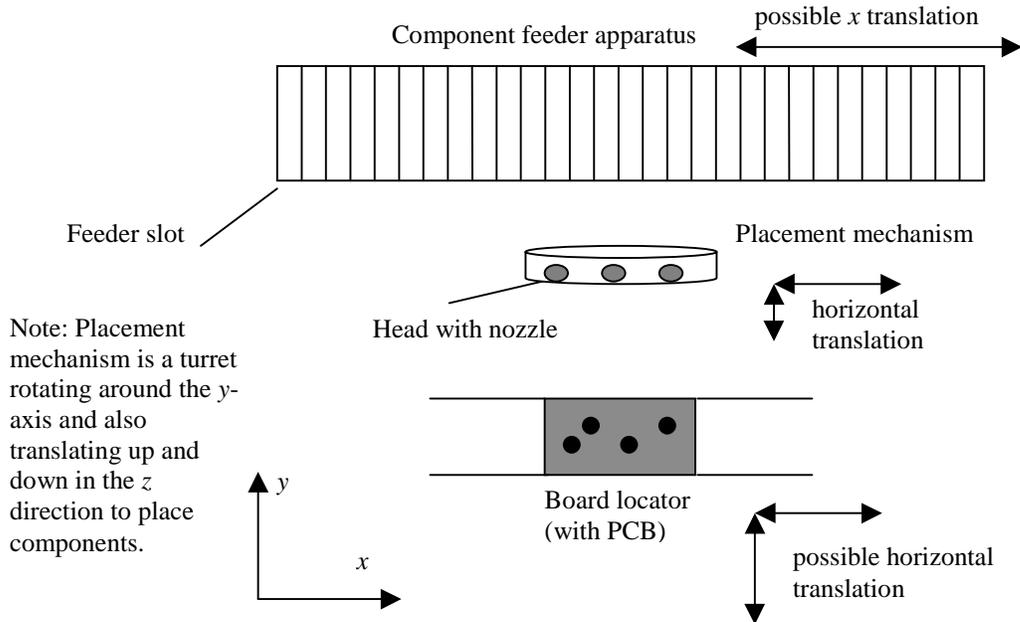


Figure 1. Schematic of placement machine mechanisms

A key factor in the operation of these machines is concurrency, which occurs when a machine performs one or more operations simultaneously [2, 9]. For example, the placement mechanism picks a set of components from the feeder, ideally loading each head with one component. In doing so, it must rotate after a pick so that the next head is positioned to pick the next component. At the same time, the placement mechanism must translate horizontally so that it is positioned at the correct feeder slot for the next component. The time between component picks, then, is the maximum of the rotation and translation times. Concurrency has a significant impact on cycle time. At the same time, it complicates cycle time analysis in terms of studying the effects of different machine configurations and operation sequences. A virtual prototype captures concurrency and other phenomena so that the analyst can focus on specifying the test conditions to be studied (e.g., machine configuration). Additionally, a virtual prototype has tremendous potential in aiding student understanding of complex machine behavior.

The remainder of this paper is organized as follows. Section 2 reviews the research literature. Section 3 describes our work in virtual prototyping. Section 4 discusses instructional use of a model. Finally, Section 5 presents conclusions and thoughts on future directions.

2. Research in Electronics Assembly

A significant body of research literature has investigated various process planning and equipment configuration problems in electronics assembly. The two main approaches taken are (i) prescriptive models that emphasize determination of a near-optimal solution to a given decision problem and (ii) descriptive models that emphasize performance evaluation of a specified set of solutions to decision problems. This section focuses on research at the machine operation level.

2.1. Decision Problems

The decision problems discussed here relate to minimization of machine cycle time. Cycle time is a function of the operations associated with moving the required components from the component feeder to the PCB and attaching them there. Two decision problems have received significant attention in the literature: the feeder arrangement problem and the placement sequencing problem [1, 4, 6, 7, 9].

In the feeder arrangement problem, one assumes that a particular machine has been assigned a set of components to be placed on a PCB. The machine also has a set of component feeder slots, to which component types may be assigned. A component type may require more than one slot, and in fact may require a non-integer number of slots. Assignment of component types to slots affects cycle time because the placement mechanism must travel between slots in performing a sequence of pick operations.

In the placement sequencing problem, one assumes that the set of components to be placed on a board is given. The problem is to determine the sequence of components to be picked and then placed on the board. The sequence impacts cycle time in two ways. First, when the placement mechanism is performing a series of placements, it must travel between various locations on the board. The sequence affects the travel time. This problem can be modeled as a special form of the travelling salesman problem [4]. Second, the sequence affects the travel time of the placement head as it moves between component feeder slots during a set of pick operations. This travel time is dependent on the feeder arrangement, as well. Hence, there is interdependence between the feeder arrangement and placement sequencing problems.

Both the feeder arrangement and placement sequencing problems are difficult when considered separately. Combined, they are even more intractable. The research literature has focused on developing optimization formulations and heuristics that apply to a specific class of machines. For example, Ahmadi *et al.* [1] and Grotzinger [7] present detailed mixed integer programming formulations for the feeder arrangement problem on machines with two component feeders and a two-head placement mechanism. Foulds and Hamacher [6] present a facility location formulation for the feeder arrangement problem and a travelling salesman formulation for the placement sequencing problem. Their model is for a non-concurrent machine. They develop a heuristic for each problem and provide an iterative procedure that uses the solution to one of the sub-problems to improve the solution to the other. The iterations continue until no significant improvement occurs. Ball and Magazine [4] model the placement sequencing as a rural postman problem and propose a heuristic that turns out to be optimal for the rectilinear distance metric. Their model applies to a non-concurrent machine.

Industry has also been actively engaged in developing formulations and heuristics for these problems. Most equipment manufacturers supply software with their machines to aid the user in configuring them. This software typically contains proprietary heuristics for problems such as placement sequencing and feeder arrangement.

2.2. Descriptive Modeling

In studying the low-level behavior and interactions of a placement machine, the most widely used methodologies for descriptive models are based on discrete-event dynamic systems theory. For example, Tirpak [11] has developed SMT Sim to study cycle time issues. SMT Sim is an object-based discrete-event simulator implemented in C. The user provides input data through a set of files that specify information about the board types to be produced and the characteristics of the component feeders. SMT Sim contains heuristics to assign component types to feeder slots and to improve the placement sequence.

Ahmadi *et al.* [2] develop a detailed emulation model of concurrency at the machine level. This model is in the form of a decision tree that specifies precedence relations between events and decisions in machine operation. Grotzinger [7] presents a Petri net model of a concurrent machine's pick and place behavior.

Discrete-event formalisms such as simulation are powerful because they allow modeling at a level of considerable detail. At the same time, though, it typically is difficult and time-consuming to develop and experiment with such a model. The approach used by Tirpak [11] is appealing because it provides a set of input data files that the analyst can manipulate without having to recode a model. This is a significant step toward building an experimental platform for virtual prototyping.

3. Virtual Prototyping

A descriptive model that allows one to assess performance of a machine or system off-line is, in a sense, a virtual prototype. In this paper, though, we refer to a virtual prototype as a descriptive model with three important characteristics:

- a detailed representation of machine behavior over time,
- a structure to the model that allows one to configure and reconfigure it easily, and
- a realistic, three-dimensional animation of machine behavior over time.

Through the detailed representation, the goal is to capture low-level machine characteristics such as acceleration and deceleration profiles and nozzle size requirements. These characteristics have an important effect on machine cycle time. Additionally, the virtual prototype model should capture the dimensions and kinematics of the machine.

By providing a structure to the model that facilitates input data corresponding to feeder arrangements and placement sequences, the intent is to create an experimental platform that can be used to prototype the machine's behavior in producing a set of boards. This model structure is important also because it is the key to integrating optimization heuristics with the simulation.

For example, these types of heuristics might serve as a front-end pre-processor to the simulation to produce the input data files.

Finally, the realistic, 3D animation is needed for two purposes. It aids in model validation, especially given the complex nature of the machine's operation. Moreover, it aids in understanding the operation of the machine itself.

3.1. Model Description

A proof-of-concept model has been developed of a Siemens 80 S20 placement machine that is installed in the Center for Board Assembly Research (CBAR) lab at Georgia Tech. The model is implemented using Virtual NC®, a simulation-based prototyping tool from Deneb Robotics. Virtual NC® is used primarily to prototype machine tool performance and to provide an off-line programming environment. It is chosen here because it provides a 3D animation of machine behavior using CAD representations, and it provides the capability to develop detailed programs to represent machine behavior in Deneb's Graphic Simulation Language (GSL), which is similar to C. A screen capture of the model is shown in Figure 2. In the figure, the component feeders are located on the right. The placement mechanism is the turret on the left. It is housed on a gantry device that provides horizontal translation (simultaneous movement in the x and y directions). It has twelve placement heads.

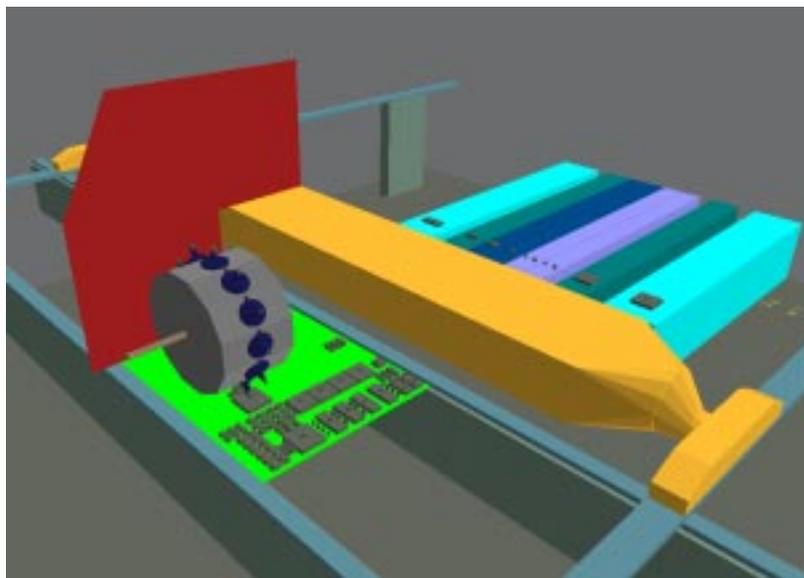


Figure 2. Placement machine model

The representation of the behavior of the machine is implemented as a GSL program. Each mechanism is defined as a device, which has an associated CAD representation. Each of these devices has axes of movement defined. For example, the placement mechanism rotates, and it also moves up and down in placing components. The nozzles on the placement heads can rotate, as well, since they may need to rotate a component to be placed on the board. The gantry device moves along rails, and within the gantry the placement mechanism is moved in the direction perpendicular to the rails. Virtual NC® provides the ability to slow down or speed up the model

execution, and it also provides the ability for the user to navigate through the CAD world to view the model from different perspectives and distances. These features are very useful in studying and understanding the machine's behavior.

3.2. Input Data Files

The simulation program reads data from files that correspond to the feeder arrangement, the component types to be placed, the board geometry and the component placement sequence. The user specifies these data files to correspond to a board production scenario to be prototyped. The input files are tables that are structured as described in Table 1.

Table 1. Input data files

File	Data
board.dat	Specifies the x and y offset locations (i.e., reference points) for each board to be produced. More than one board can be produced at a time. Each board to be produced has an entry in the table, and each entry specifies an x and y location.
chips.dat	Specifies the component types. Each component type has an entry in the table. Each entry specifies a unique identifier for the component type.
feeder.dat	Specifies the configuration of the feeders. Each feeder has an entry in the table. Each entry specifies the x , y and z coordinates of the location from which a chip is picked, and the component type that is housed in the feeder.
recipe.dat	Specifies the placement sequence. Each entry in the table corresponds to a placement, and the order in the table corresponds to the placement sequence. Each entry specifies the x , y and z coordinates of the placement location on the board, the rotation angle of the placement (if any), and the feeder slot from which the component is to be picked.

3.3. Validation

Model validation is an important issue in virtual prototyping. Given the detailed nature of the data needed to represent machine behavior, it is a challenging task. At present, the placement machine model needs further validation against the real machine. This will involve extensive data gathering and comparison between model cycle time and machine cycle time for test sets of boards.

4. Instructional Use of the Model

In recent years, there has been a growing interest in improving the state-of-the-art in instructional techniques and computational tools for engineering education (e.g., [5, 8, 10]). Trends include active learning, use of computer simulation to emulate real engineered systems, and group learning exercises. In the winter term of 1998, the placement machine model was used in Electronics Assembly Systems, a graduate level course offered by the School of Industrial and Systems Engineering at Georgia Tech. This course is part of the multidisciplinary Computer Integrated Manufacturing Systems program; hence, students from Mechanical Engineering were enrolled, as well. Total enrollment for the class was 12 students.

The students were given a two-part assignment using a baseline model of the Siemens placement machine. This baseline model specified a four identical boards to be produced simultaneously (components and board geometry) and an initial component placement sequence and feeder arrangement. In the first part of the assignment, the students were asked to do the following:

- run the model to determine the cycle time for a single set of boards,
- develop a Gantt-chart representation and a mathematical expression of the most efficient cycle time, and
- develop a heuristic to improve the cycle time by modifying the placement sequence.

In the second part of the assignment, they were asked to do the following:

- determine a new feeder arrangement to improve the cycle time (without changing the placement sequence), and
- share their thoughts on how to solve the combined problem of placement sequencing and feeder arrangement.

During the course of the assignment, the students were actively engaged in improving the cycle time of the model by manipulating the input data files. The students were allowed to work in groups of two, and all the groups were encouraged to share their best results to date with the other groups using e-mail. This led to lively exchanges at several points when students had to defend their results. Feedback from the instructor and students indicates the following [3].

- Compared to previous offerings of the course, the virtual prototype model significantly improved the students' ability to model a placement machine and develop Gantt charts for its operations.
- It also significantly improved their ability to look for opportunities to optimize cycle time and to understand improvement heuristics.
- In one instance, it helped the students understand some counter-intuitive results. Two heuristics were applied to the placement sequencing problem. The one that should have given worse performance actually gave better performance due to a specific characteristic of the machine. The model helped the students understand why this occurred.
- Having the students work in groups and communicate proved very useful because the students had to justify their results to the other students, who sometimes were very skeptical.

In addition, the students were able to point out shortcomings of the model that will be the basis for further improvement. Clearly, at this point the data is anecdotal on instructional use of the model. However, it is encouraging. We plan to deploy this type of model in other courses, and we plan to develop more formal metrics by which the use of this and similar models can be evaluated for instructional effectiveness.

5. Conclusions and Future Work

This paper has presented a virtual prototyping methodology for placement machines in electronics assembly. Virtual prototyping models can be used to predict cycle time performance for given machine configurations and placement sequences. In addition, these models have

tremendous potential to aid in teaching students about important issues in electronic assembly systems.

Plans for future work include (i) validating the model rigorously against the existing machine, (ii) refining the model and associated assignments for future use in courses, and (iii) developing and implementing optimization formulations and heuristics that will generate the input files for the simulation, and then integrating the optimization and simulation models.

Acknowledgements

The authors gratefully acknowledge support from the W. M. Keck Foundation. The authors also wish to thank Prof. Jane Ammons at the Georgia Institute of Technology and the students in her Electronics Assembly Systems course for their assistance, encouragement and feedback.

References

- [1] Ahmadi, I., S. Grotzinger and D. Johnson (1988). "Component Allocation and Partitioning for a Dual Delivery Placement Machine," *Operations Research*, Vol. 36, No. 2, pp.176-191.
- [2] Ahmadi, I., S. Grotzinger and D. Johnson (1991). "Emulation of Concurrency in Printed Circuit Card Assembly Machines," *International Journal of Flexible Manufacturing Systems*, Vol. 3, No. 1, pp. 45-70.
- [3] Ammons, J. C. (1998). Personal communication.
- [4] Ball, M. O. and M. J. Magazine (1988). "Sequencing of Insertions in Printed Circuit Card Assembly," *Operations Research*, Vol. 36, No. 2, pp. 192-201.
- [5] Dessouky, M. M., D. E. Bailey, S. Verma, S. Adiga, G. A. Bekey and E. J. Kazlauskas (1998). "A Virtual Factory Teaching System in Support of Manufacturing Education," working paper, Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA.
- [6] Foulds, L. R. and H. W. Hamacher (1993). "Optimal Bin Location and Sequencing in Printed Circuit Board Assembly," *European Journal of Operational Research*, Vol. 66, No. 3, pp. 279-290.
- [7] Grotzinger, S. (1992). "Feeder Assignment Models for Concurrent Placement Machines," *IIE Transactions*, Vol. 24, No. 4, pp. 31-46.
- [8] Lu, G. B., M. Oveissi, D. Eckard and G. W. Rubloff (1996). "Education in Semiconductor Manufacturing Processes Through Physically-Based Dynamic Simulation," *Proceedings of the 1996 Frontiers in Education Conference*, Salt Lake City, UT, pp. 250-253.
- [9] McGinnis, L. F., J. C. Ammons, M. Carlyle, L. Cranmer, G. W. Depuy, K. P. Ellis, C. A. Tovey and H. Xu (1992). "Automated Process Planning for Printed Circuit Card Assembly," *IIE Transactions*, Vol. 24, No. 4, pp. 18-30.
- [10] Rosetti, M. D. (1997). "Activate This Classroom at Time Now," *Proceedings of the 1997 Winter Simulation Conference*, Atlanta, GA, pp. 1383-1389.
- [11] Tirpak, T. M. (1993). "Simulation Software for Surface Mount Assembly," *Proceedings of the 1993 Winter Simulation Conference*, Los Angeles, CA, pp. 796-803.