ELSEVIER
Discrete Applied Mathematics 123 (2002) 339–361
DISCRETE APPLIED MATHEMATICS

Production planning problems in printed circuit board assembly

Yves Crama[a,*], Joris van de Klundert[b], Frits C.R. Spieksma[b]

[a] *Ecole d'Administration des Affaires, University of Liège, Boulevard du Rectorat 7 (B31), 4000 Liège, Belgium*
[b] *Department of Applied Economics, Katholieke Universiteit Leuven, Naamsestraat 69, B-3000 Leuven, Belgium*

Received 5 January 2000; accepted 16 October 2000

Abstract

This survey describes some of the main optimization problems arising in the context of production planning for the assembly of printed circuit boards. The discussion is structured around a hierarchical decomposition of the planning process into distinct optimization subproblems, addressing issues such as the assignment of board types to machine groups, the allocation of component feeders to individual machines, the determination of optimal production sequences, etc. The paper reviews the literature on this topic with an emphasis on the most recent developments, on the fundamental structure of the mathematical models and on the relation between these models and some 'environmental' variables such as the layout of the shop or the product mix. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Production planning; Scheduling; Sequencing; Printed circuit boards; Integer programming models

1. Introduction, facts and figures

The assembly of printed circuit boards (PCBs) has generated a huge amount of industrial activity over the last 20 years. PCBs are consumed as inputs by three major industrial sectors: computers, telecommunications and consumer electronics represented 72.5% of the total consumption in 1998 [47]. Although it seems difficult to gather precise figures, Nakahara [47] indicates that world PCB production grew by 5% in

* Corresponding author. Tel.: +32-4-366-3077; fax: +32-4-366-2821.
E-mail addresses: y.crama@ulg.ac.be (Y. Crama), j.vandeklundert@math.unimaas.nl (J. van de Klundert), frits.spieksma@econ.kuleuven.ac.be (F.C.R. Spieksma).

PII: S0166-218X(01)00345-6

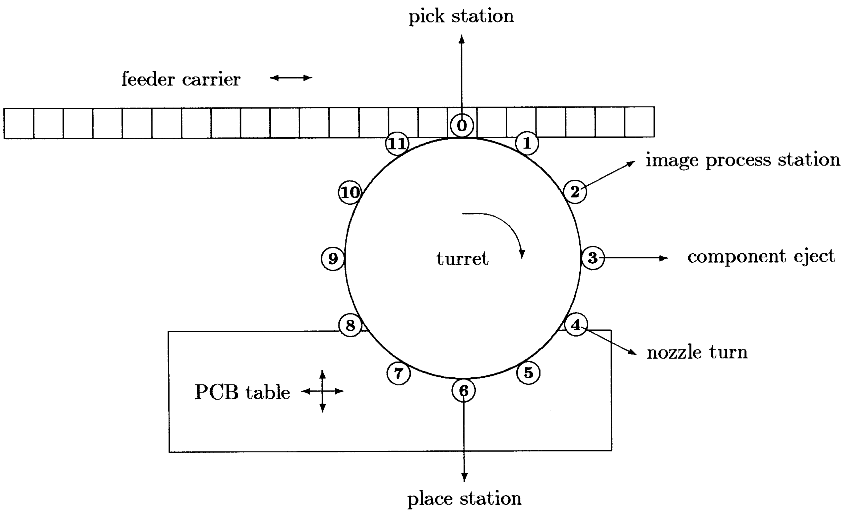

Fig. 1. A machine of the Fuji CP family.

1998, to a total value of roughly \$35 billion. The top 15 countries accounted for 92% of this worth, with Japan and the USA producing more than 50% of the total output.

Over the years, PCB production has evolved from a labor-intensive activity to a highly automated one, characterized by steady innovations at the level of design and manufacturing processes. Nowadays, programmed automation has gained the upper-hand in assembly operations. In their description of benchmark PCB assembly factories, Mody et al. [43] estimate that, in industrialized countries, a typical shop features 25–30 machines, for a total equipment value exceeding \$1.5 million.

These sophisticated machines perform a large number of high speed, high precision assembly operations requiring various tools and components. Some operating features of the Fuji CPII placement machines are mentioned for instance by Bard et al. [12] (see Section 2 and Fig. 1 for the terminology): turret rotation speed of one station per 0.15 s; table movement speed of 20 mm per 0.15 s; feeder carrier speed of one slot per 0.15 s; duration of picking or placement actions: 0.10 s; placement rate of over 12,000 components/h; error rate of less than 1 in 10,000.

The competition faced by PCB manufacturers creates a need for production efficiency which is achieved—depending on the specific market—by assembling either a few product types in large volumes or a large variety of products in small volumes. Jain et al. [32] compared three of Hewlett–Packard's production sites and describe production characteristics ranging from low mix (less than 20 board types) high volume operation (batches of more than 100 units) to high mix (150 board types) low volume operation (batches of 10 to 25 boards). A detailed discussion of manufacturing flexibility in PCB assembly is provided by Suarez et al. [53], who mention a plant producing only two board models and another one producing more than 2000 different models! The plant studied by Feo et al. [24] assembles 20,000–80,000 boards/month, but Mody et al. [43] consider an output of 40,000 boards/year to be more typical.

All the above features interfere with numerous constraints and conflicting managerial objectives to pose challenging production planning problems. In fact, in the conclusions of their study, Mody et al. [43] point out that PCB manufacturers, both in less developed countries and in newly industrialized countries, will need (among other factors) to increase process efficiency and to master production planning and control in order to improve their competitive situation.

In order to cope effectively with such requirements, decision support systems based on specialized planning and scheduling models may prove a major asset for PCB producers. Many researchers have investigated such models for PCB assembly and have published numerous papers on this topic in the operations research, industrial engineering and production management literature. We are going to review some of this literature, with an emphasis on the most recent developments, on the fundamental structure of the mathematical models and on the relation between these models and some 'environmental' variables such as the layout of the shop or the product mix, with the hope and ambition to provide useful guidance to the reader. For complementary viewpoints or additional information, we refer the reader to excellent previous surveys by McGinnis et al. [43] or Ahmadi [1]. Extensive bibliographic references can also be found on several Internet sites: http://www.econ.kuleuven.ac.be/tew/academic/kwantmet/members/frits/Bibliography/bibliogr.htm (Crama, van de Klundert and Spieksma), http://www.Fabtime.com/library.htm (Robinson), http://www.eas.asu.edu/∼masml (Fowler and Runger), and http://www.cs.utu.fi/scheduling/Default.htm (Nevalainen et al.).

2. Generic assembly process

Before discussing the fundamental issues involved in the PCB production planning process, it is necessary to give a description of the generic steps involved in the assembly of a printed circuit board.

For our purpose, PCB assembly consists in placing (inserting, mounting) a number of electronic components of prespecified types at prespecified locations on a bare board. Several hundred components of a few distinct types (resistors, capacitors, transistors, integrated circuits, etc.) may be placed on each board.

An automated PCB shop involves several computerized machines (or workstations), possibly with different characteristics, which take care of the assembly operations (see e.g. [24] for a pictorial representation of such a shop). The stations may be linked by a material handling system which allows for some flexibility in routing the boards through the shop. In this case, we will say that the shop is a *flexible* or *decoupled cell*. Most often, however, the machines are laid out into distinct *assembly lines*, or *coupled systems*, and a conveyor connects the machines within each line.

As already mentioned, the placement machines may be of various types. From the point of view of the operations researcher, this is somewhat unfortunate, since the technological characteristics of the equipment influences the nature of some of the planning problems to be solved and the formulation of the associated models. We will have opportunities to return to this point. For the time being, let us settle for a generic description of the placement machines (see e.g. [42] or [23] for more details).

Each machine essentially consists of a *worktable*, a *feeder carrier* (or magazine, or rack) and a *pick-and-place device* (see Fig. 1 for an example). The worktable holds the PCB during the placement operations. Depending on the machine, the table can either be stationary or mobile in the X–Y plane. The components to be placed on the PCB are released by *component feeders* which have to be loaded into the slots of the carrier prior to production. Usually, the carrier can move by translation along an X-axis. Finally, the pick-and-place device allows to retrieve each component from the appropriate feeder and to place it on the board. Very different designs and operating modes exist for the pick-and-place device. Sometimes, it can only move in the Y–Z plane (see e.g. [38]). In other cases, it features 12 workheads arranged circularly on a turret: in each pick-and-place operation, head 0 picks a component while head 6 places another one; thereafter, the device rotates by 30° and a similar operation is repeated (see Fig. 1 from [12,18]). Yet other types of designs are described by Ball and Magazine [10], Ahmadi et al. [3], Leipälä and Nevalainen [37], Crama et al. [19], van Laarhoven and Zijm [35], Francis et al. [26], etc.

McGinnis et al. [42] use the term *machine cycle* to designate a series of consecutive operations beginning with a component retrieval, ending with a component placement and consisting of only one retrieval and one placement. This allows to classify placement machines into two major categories: *sequential* machines are those for which each machine cycle involves exactly one component (the same component is gripped and immediately placed) while *concurrent* machines are those for which each cycle involves the retrieval of one component and the placement of a previously retrieved component (concurrent machines may perform several operations simultaneously). The Fuji CP machine illustrated in Fig. 1 is a concurrent machine.

3. Planning hierarchy

Production planning decisions are frequently formulated in a hierarchical framework where they decompose into long term (strategic), medium term (tactical) and short term (operational) issues. There remains quite a lot of freedom, however, as to the 'best' decomposition to be used in a given situation. The answer to this question depends, among others, on

- characteristics of the product mix (diversity of PCB types, batch sizes, etc.),
- characteristics of the equipment (layout, number of machines, details of the operating mode, etc.),
- managerial policy regarding for instance the frequency of setups or the willingness to redesign the lines on a regular basis.

See e.g. [24] for a global vantage point on the planning process. It should be noted that very similar issues come up in the management of flexible manufacturing systems; see e.g. [50] and a comparison of PCB and FMS environment in Ammons et al. [6].

In this paper, we consider the long-term decisions to be given and we concentrate on tactical and operational decisions. In particular, we assume the *demand mix* and the *shop layout* to be fixed exogenously. Under these conditions, the production planning process must (at least) address the following list of subproblems SP1 to SP8. It must determine:

SP1. an *assignment* of PCB types to product families and to machine groups (cells or lines);
SP2. an *allocation* of component feeders to machines;
SP3. for each PCB type, a *partition* of the set of component locations on this board type, indicating which components are going to be placed by each machine;
SP4. for each machine group, a *sequence* of the PCB types, indicating in which order the board types will be produced on these machines;
SP5. for each machine, the *location of feeders* on the carrier;
SP6. for each pair consisting of a machine and a PCB type, a *component placement sequence*, that is a sequence of the placement operations to be performed by the machine on this board type;
SP7. for each pair consisting of a machine and a PCB type, a *component retrieval plan*, that is, for each component on the board, a rule indicating from which feeder this component should be retrieved;
SP8. for each pair consisting of a machine and a PCB type, a *motion control specification*, that is, for each component, a specification of where the pick-and-place device should be located when it picks or places the component.

(Alternative hierarchical decomposition schemes have been proposed by various authors; see e.g. [1,43,56], etc.)

Observe that problem SP1 is posed at the level of the whole assembly shop and involves all products to be assembled, SP2–SP4 usually arise for each product family at the level of assembly lines or cells, and SP5–SP8 deal with individual machines.

Decisions SP1–SP8 must be made in such a way as to optimize some criterion of production performance. The criterion which is most commonly considered in the literature is *makespan minimization* or, in the context of repetitive assembly, *cycle time minimization*. Other criteria may also be of importance, but are less frequently tackled; for instance, van Zante-de Fokkert and de Kok [56] formulate a variant of SP1 with the objective to minimize the sum of assembly, setup and inventory holding costs.

The above list of decisions covers a wide variety of situations. In any specific one, however, some of the subproblems may become vacuous. For instance, it is quite common to assume that only one feeder is available for each type of component (due to the inventory costs of components). In such a case, subproblems SP3 and SP7 vanish altogether: indeed, subproblem SP3 only arises when a same feeder type is loaded on several machines and subproblem SP7 only arises when a same feeder type is loaded in several slots of a machine.

On the other hand, a host of operational details may encumber the description of the fundamental planning decisions and are frequently omitted in the literature. Some of these details could easily be taken into account, as they only affect the value of certain parameters of the models (for instance, the speed of the pick and place device may depend on the type of the components that it carries). Others, however, may have a significant impact on the formulation and on the complexity of the optimization models (for instance, long translations of the feeder carrier are to be avoided, as they are responsible for additional shocks and wear of the carrier; some feeders may occupy more than one slot on the carrier; etc.).

All in all, however, the major difficulty with the list SP1–SP8 is that all its subproblems are tightly intertwined. This fact has been underlined by virtually all researchers in the field (see e.g. [1]). Not only does the formulation of any subproblem heavily depend on the solution computed for problems of *higher* level, but it also depends, in a very significant way, on the solution of problems of *lower level*. This is true, of course, of any hierarchical decomposition scheme, but appears to be especially troublesome in the present case. As a consequence, several authors have adopted solution procedures which iterate between subproblems, rather than one-pass procedures through the list of decisions.

In this survey, for the ease of exposition, we are going to tackle problems SP1–SP8 in reverse order, starting from detailed scheduling questions to finish with the more encompassing (and arguably, more crucial) tactical questions. Thus, we are successively going to consider single machine single product problems (Section 4), then single machine multi-product problems (Section 5), before we turn to the more realistic multi-machine, multi-product environment and a discussion of issues surrounding setup decisions (Section 6).

4. Single machine, single board type problems

Let us first consider the case where a single PCB type must be repeatedly assembled on a single machine, with the objective of makespan (or cycle time) minimization. In this case, the only subproblems to be solved are:

SP5. feeder location;
SP6. placement sequencing;
SP7. component retrieval;
SP8. motion control.

Van Laarhoven and Zijm [35] emphasize the fact that the latter decisions (as opposed to other planning and scheduling decisions) are directly relevant to the *production preparation* function, which leads to the specification of the numerical control programs guiding the assembly operations for each particular PCB.

Let us now discuss each of these problems in turn, starting at the ‘bottom’ of the hierarchy.

4.1. Motion control (SP8)

Suppose that feeder locations have been determined, that a component placement sequence is given and that it is known for each location where the component to be placed must be retrieved from (that is, a component retrieval plan is known). In this situation, there may still remain one decision left to make: for placement machines that feature a pick-and place-device that can move in the X–Y plane, as well as a rack and a table that can move in the X-direction, one must determine where the device meets the rack (resp. the board) to pick (resp. to place) the appropriate components. Greedy approaches that avoid waiting times for the pick-and-place device are suggested in Su et al. [52] and Wang et al. [59]. These studies also demonstrate the potential makespan gain when allowing non-static pick and place points versus static ones and try to compute placement sequences and feeder locations (see Subsections 4.3) that minimize makespan.

4.2. Component retrieval (SP7)

Assume now that feeder locations and a component placement sequence have been determined. If several component feeders of a same type have been assigned to more than one carrier slot, it becomes necessary to decide from which feeder each component should be retrieved. Of course, different decisions for a specific component may result in different assembly makespans for the board. This issue is raised by Bard et al. [12] for the Fuji CPII machine (see Fig. 1) and is further investigated by Crama et al. [17]. It is also briefly mentioned by Ahmadi et al. [2].

The complexity of the component retrieval problem depends very much on the modus operandi of the placement machines. For most sequential machines, it can be modeled and solved as a shortest path problem. The same holds true for the Fuji CPII machine if the start of a pick activity coincides with the start of a place activity. However, the problem becomes much less trivial when we lift this (restrictive) assumption. Crama et al. [17] show that the problem can still be solved in polynomial time by dynamic programming, but that a slight generalization is already NP-hard.

4.3. Feeder location and placement sequencing (SP5 and SP6)

Starting with [22], numerous researchers have investigated the joint problem of feeder location and placement sequencing. Let us sketch a formulation of this problem for a sequential machine. We let n denote the number of components to be placed, $f(i)$ denote the feeder delivering component i ($i = 1, \dots, n$) and C denote the number of slots available in the rack. The 0–1 decision variables are

$x_{ij} = 1$ iff component j is placed directly after component i ($i, j = 1, \dots, n$),

$y_{f(i),s} = 1$ iff a feeder for component i is stored in slot s ($i = 1, \dots, n$, $s = 1, \dots, C$).

Using these variables we can write down the following model:

$$\text{minimize} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{s=1}^{C} c_{ijs} x_{ij} y_{f(j),s} \tag{1}$$

$$\text{s.t. } x \text{ describes a Hamiltonian path,} \tag{2}$$

$$y \text{ describes a feasible assignment,} \tag{3}$$

where c_{ijs} denotes the time elapsed between placing component i and placing component j when the feeder $f(j)$ is stored in slot s. For any fixed assignment of feeders to carrier slots, the placement sequencing problem is (essentially) a traveling salesman problem or shortest Hamiltonian path problem (this is true for sequential as well as for concurrent machines). It is easy to understand, however, that the 'distance' or travel time between successive placements is influenced by the location of the feeders, since a 'pick' operation takes place between successive insertions. Conversely, given any sequence of placement operations, the feeder location problem displays the structure of a linear (or, for some types of machines, quadratic) assignment problem, where the 'cost' of assigning a feeder to a particular slot depends on the movements to be performed to and from this slot. Alternatively, the feeder location problem can also be modeled as a facility location problem.

These observations motivate a popular algorithmic approach which consists in tackling both problems simultaneously by iterating between (heuristic) solutions of the feeder location problem and the placement sequencing problem. This approach was initiated (in another manufacturing framework) by Walas and Askin [58] and was also used by Leipälä and Nevalainen [37] or by Broad et al. [13] for PANASERT machines, by Crama et al. [19] for CSM-60 placement machines, by Egbelu et al. [23], Foulds and Hamacher [25], Leon and Peters [38], Moyer and Gupta [45], etc. Recently Altinkemer et al. [5] have proposed an integrated model and an algorithm which reduces the solution of (SP5)–(SP6) to a number of vehicle routing problems. If the vehicle routing subproblems are solved within an ε-error guarantee, then the same guarantee holds for the integrated model.

In order to conduct a finer analysis of the theoretical properties of the models, some authors have rather elected to focus on one of the two subproblems: they explicitly assume to have a solution of one of the two problems and investigate the properties of the second one. Ahmadi et al. [2], for instance, consider the feeder location problem for the DYNAPERT placement machine, *given* a component placement sequence (the placement sequence could arise in the course of the iterative procedures mentioned above, or could be obtained by simple traveling salesman heuristics like those described by Gaboune et al. [27]). They show that, in their setting, the feeder location problem is NP-hard and they provide an approximation algorithm with worst-case ratio $\frac{3}{2}$. Bard et al. [12] address a similar problem for the Fuji CPII. They propose a quadratic integer programming formulation which they attack by Lagrangian relaxation techniques. Moyer and Gupta [44] or Dikos et al. [21] also treat the component placement sequence as an input.

Conversely Drezner and Nof [22], Ball and Magazine [10] or van Laarhoven and Zijm [35] assume that the feeder location problem has been computed first (by solving a linear assignment model in which the total placement time of all the components retrieved from a given feeder is roughly approximated). For known feeder locations, the placement sequence problem can then be tackled in a second phase.

Notice that, even for fixed feeder locations, modeling the elapsed time between two successive placements may not be entirely straightforward. Independently of the physical distance between such successive placements, the elapsed time is clearly limited from below by the time required to carry out a series of unavoidable operations (e.g., for Fuji CP machines: pick a component, rotate the turret by 30°, move the feeder carrier, and so on). This gives rise to so-called 'free' movements, whose execution time is 'masked' by the execution time of unavoidable operations. For concurrent machines, in particular, this results in complex 'distance metrics' in the formulations of the placement sequencing problem, but also raises opportunities for improved sequencing. These aspects are discussed by Ahmadi et al. [2], Ahmadi et al. [3], Bard et al. [12], Crama et al. [18], Egbelu et al. [23], Grotzinger [29], etc.

In simpler cases, the special structure of the distance metrics can sometimes be exploited to derive tailor-made heuristics (see [10,26], etc.). Vicziàn [57] shows that the algorithm proposed in [26] has worst-case ratio equal to $\frac{3}{2}$. Van Laarhoven and Zijm [35] use a simulated annealing heuristic to compute a near-optimal placement sequence.

Finally, observe that, if several feeders of a same type have been assigned to the machine, then the formulation of the placement sequencing problem becomes somewhat tricky. Indeed, the 'distance' between successive placements is now influenced by the solution of the component retrieval subproblem… which we solved (in Section 4.2) under the assumption that the component placement sequence was known! To get around this difficulty, Crama et al. [18] solve the placement sequencing subproblem by an exchange heuristic in which the component retrieval plan is kept fixed over a number of successive iterations and reoptimized once in a while.

5. Single machine, multiple board type problems

As we will see below, a placement machine may frequently be setup for a family of boards (*family setup*, see Section 6), rather than for a unique board type. In such a case, the feeder location problem must be solved simultaneously for all boards in the family, as opposed to placement sequencing which can be solved anew, and independently, for each board type. Thus, there arises an obvious asymmetry between the two subproblems and some of the approaches mentioned in the previous section may become less manageable.

In this multiple-board setting, the feeder location problem can be viewed as follows: we want to

$$\begin{array}{ll} \text{minimize} & \mathit{makespan}(\varphi) \\ \text{s.t.} & \varphi \text{ is a feasible feeder assignment,} \end{array}$$

where *makespan*(φ) is a very complex function of the assignment φ, since it depends on the solution of the placement sequencing problem for *all* boards in the family. The literature on this problem is extremely scarce. As in the single-board version, it is possible to use iterative heuristics which alternate between the computation of tentative feeder assignments and of placement sequences for all board types. This approach is described in [42,38]. Notice, however, that it may involve the solution of a large number of traveling salesman problems. For instance, with three machines and nine board types (as in [18]), 27 traveling salesman problems must be solved for each feeder assignment. If local search is used in order to improve the location of feeders, then the number of TSP instances may grow very large.

In order to reduce the computational burden of the procedure, Crama et al. [18] suggest to rely on a very fast approximation of the objective function *makespan*(φ), which can be used for optimizing feeder locations by local search. In their experiments, the approximation accelerates the search and proves quite accurate.

Dikos et al. [21] develop a genetic algorithm for the feeder location problem with multiple board types, under the assumption that placement sequences are known in advance.

There does not seem to be much more work on the multi-board version of the feeder location and placement sequencing problems: in view of the practical relevance of these problems, there is here ample opportunity for further research.

6. Multiple machines: setup policies

When more than one board type is to be produced over the planning horizon, a policy has to be adopted regarding the conditions under which new feeder setups can be performed. A feeder setup may affect the allocation of component feeders to the machines as well as the location of feeders on the carriers (cf. problems SP2 and SP5 in Section 3). Observe however that, because of interdependencies between the various subproblems, setup policy actually encompasses a broader set of issues, partially reflected in problems SP1–SP5. The practical importance of setup policies cannot be overestimated: Jain et al. [32] mention for instance that, at some Hewlett–Packard shops, over 50% of the production time is spent in setups.

Several types of setup policies have been identified in the PCB literature (see e.g. [6,9,32,42,38] etc. Notice that similar distinctions have also been established in the literature on tool management for flexible manufacturing systems; see e.g. [16,28,49,51]). For a given family of board types to be produced over the planning horizon, a possible typology of setup policies goes as follows:

(a) *tear-down* setups [32] (also called *single unique setup* [42] or *complete setup* [9]): between the assembly of successive board types, all feeders are removed and a new setup is performed;
(b) *partial* setups [9,38]: the removal and replacement of feeders is allowed between successive board types; there are several variants of this idea, to be discussed in Section 6.3;
(c) *family setups* [42]: no feeder setup is allowed between successive boards in the family; thus, the assembly line (or cell) must have sufficient carrier capacity to accommodate all the feeders required by the family.

Ammons et al. [6] provide a nice review of setup policies in connection with machine grouping, product grouping and component allocation issues. We would like to emphasize here that the setup policy adopted by a plant is, to a large extent, influenced by its product mix (which we assumed earlier to be exogenously given). In the sequel, we will refine the formulation of problems SP1–SP8 under different setup hypotheses.

For the ease of exposition, we start the discussion with the most clear-cut situations, i.e. tear-down policy (Section 6.1) and family setups (Section 6.2), and we finish with the more complex case of partial setups (Section 6.3).

6.1. Tear-down policy

Consider first the tear-down policy. This policy appears to be most adequate when the product mix displays a small variety of PCB types, assembled in relatively large batches. In this case, the high setup times incurred under the tear-down policy can be offset by the productivity gains resulting from customized feeder allocation and location decisions.

Under the tear-down policy, most of the planning hierarchy collapses to a collection of simpler questions bearing on a single board type. Essentially, the tear-down policy reduces the planning problem to a single board multiple machine situation. For instance, the issue of PCB sequencing (SP4) vanishes and the feeder location problem (SP5) is solved anew for each PCB type.

The major remaining decisions concern the allocation of feeders and of placement operations to machines, i.e. SP2 and SP3. For an assembly line, the most appropriate model formulation requires to allocate the feeders and the operations so as to minimize (an estimate of) the workload of the bottleneck machine. Such models have been used, for instance, by Crama et al. [19] or van Laarhoven and Zijm [35] for a single PCB type, i.e. in a tear-down policy framework. We will come back to such models in Section 6.2, for multiple board types. Once these problems have been solved, the remaining problems (feeder location, placement sequencing and component retrieval and motion control specification, viz. subproblems SP5–SP8) are single machine problems that have already been discussed in Section 4.

6.2. Family setups

Family setups appear adequate when there is a high (to medium) variety of PCB types, assembled in small (to medium) batches. Indeed, in such situation, the assembly time to be gained from improved feeder allocation/location for each individual board type may not compensate for additional setup time. Some plant managers also prefer to avoid frequent setups which may easily lead to human errors, and thus, to quality and/or productivity losses.

In practice, family setups may actually arise in (at least) two different frameworks. In both cases, we may assume that, prior to the start of the planning horizon, the PCBs to be produced over the given horizon have been partitioned into families (possibly, a unique family). Then,

- either each family is assigned to a distinct group of machines (assembly line or workcell) and each group is setup once for the assembly of the whole family;
- or the families are successively produced on the same line (or in the same workcell) and a new setup is performed before the production of each family.

According to the typology presented above, the second situation should be classified in the category of ‘partial setups’, but it shares in fact all the characteristics of family setups. In particular, the question that naturally arises in both cases is (cf. SP1): how to assign PCB types to product families and—in the first case—to machine groups?

6.2.1. Assignment of PCB types to product families and to machine groups (SP1)

Assigning PCB types to product families is a decision very much akin to those considered in the group technology (GT) literature on 'cell formation' or in the FMS literature on 'job grouping' (see e.g. [14,16,50,55]).

In the GT framework, products are grouped by a clustering algorithm based on component commonality between boards. The 'capacity' of the feeder carriers is not directly taken into account by classical clustering procedures, which must therefore be adapted in an ad hoc fashion; see e.g. [46] for an illustration.

The FMS job grouping model on the other hand, explicitly takes the carrier capacity into account. In its best known version, the objective function of this problem attempts to minimize the number of families to be formed. This model has been extensively studied, both from a computational and from a theoretical point of view (see [55,20] and the survey in [16]). It provides a reasonable proxy of the makespan minimization problem when all the families have to be produced on a single line of machines and when the setup time strongly dominates the assembly time.

By contrast, in the multi-line (or multi-cell) setting, the number of machine groups is fixed a priori. Hence, a more adequate formulation of SP1 concentrates on the allocation of product types to machine groups so as to minimize the workload of the most heavily loaded machine group (here, a product family is defined as the collection of PCB types assigned to a same machine group). The resulting model is akin to bin packing or parallel machine scheduling models. In order to formulate SP1 as an integer programming problem, let $i=1,\ldots,I$ denote the available machine groups, let $k=1,\ldots,K$ denote PCB types, let $j=1,\ldots,J$ denote the feeders to be used, let a_{ik} be the estimated assembly time for all boards of type k on machine group i, let N_i be the total (aggregated) capacity of all feeder carriers of the machines in group i and let δ_{jk} be a 0–1 parameter which takes value 1 if PCB type k requires feeder j and value 0 otherwise. The 0–1 decision variables are

$$y_{ik}=1 \quad \text{if board type } k \text{ is assigned to machine group } i,$$
$$z_{ij}=1 \quad \text{if feeder } j \text{ is set up on machine group } i$$

and the model can be written as

$$\text{minimize} \quad \max_{i=1,\ldots,I} \sum_{k=1}^{K} a_{ik}\, y_{ik} \tag{4}$$

$$\text{s.t.} \quad \sum_{i=1}^{I} y_{ik} = 1 \quad \text{for all } k, \tag{5}$$

$$\sum_{j=1}^{J} z_{ij} \leqslant N_i \quad \text{for all } i, \tag{6}$$

$$\delta_{jk}\, y_{ik} \leqslant z_{ij} \quad \text{for all } i,j,k, \tag{7}$$

$$y_{ik} \in \{0,1\} \quad \text{for all } i,k, \tag{8}$$

$$z_{ij} \in \{0,1\} \quad \text{for all } i,j. \tag{9}$$

A distinguishing feature of the above model is that the machine groups are viewed as completely decoupled (each product type is processed by exactly one group—see constraint (5)), in agreement with the layout and the organization of many assembly shops. Moreover, the model differs from feeder allocation (SP2) or feeder location (SP5) models since it assigns feeders to groups of machines, rather than to individual machines or individual slots, and since it treats feeder capacity at an aggregated level only (constraint (6)).

This type of integer programming model has not been widely studied in the literature. Hillier and Brandeau [31] propose a model (BIP4) which is very similar to (4)–(9), except that its objective is to minimize total assembly cost (or time) rather than to balance the workload. They develop an exact algorithm and a heuristic based on Lagrangian relaxation. In a more general model (where partial setups are allowed), Balakrishnan and Vanderbeck [9] propose to minimize the setup cost, but add an upper-bound on the allowed workload per machine group (so, when restricted to family setups, their model is essentially equivalent to (4)–(9); see Section 6.3 for more details). They attack this model by column generation techniques. Finally, it should be noted that model (4)–(9) shares very obvious similarities with some of the integer programming models proposed for the job grouping problem in the FMS literature (see e.g. [20]).

A difficulty with the above model is that the total assembly time (a_{ik}) is very difficult to estimate, since it depends in a complex way on the set of PCBs which are allocated to each machine group and thus, on the solution of remaining subproblems in the list SP1–SP8.

To proceed, let us now assume that there is a unique family of boards to be produced by an assembly line or cell (i.e., let us assume that the family formation problem has been solved) and let us turn to the remaining subproblems.

6.2.2. Feeder allocation for assembly lines (SP2 and SP3)

Consider a single assembly line which is to be set up (once) for the production of a family of PCB types, say types $1,\ldots,K$. In this setting, it is usually assumed that production takes place in *batch mode*, where batch k consists of d_k boards of type $k=1,\ldots,K$. Provided all batch sizes are moderately large, this implies that the issue of PCB sequencing (SP4) can be disregarded altogether, as it will not affect performance in a significant way. The remaining issues to be addressed concern the feeder allocation problem (SP2) and, if relevant, the auxiliary problem SP3 (recall that SP3 only arises if feeders containing a same component type have been assigned to several machines). Then, once SP2 and SP3 have been solved, the planning problem is reduced to a collection of single machine single board subproblems (one for each machine in the line), as in Section 4.

McGinnis et al. [42] suggest that, for SP2–SP3, the most appropriate objective function consists in minimizing the sum over all board types of the makespans of these board types on their bottleneck machines. Of course, different types of PCBs, and therefore different batches, may have different bottleneck machines. For simplicity, let us restrict our attention to the feeder allocation problem (SP2) by assuming that each component feeder can only be used once. Let $t_{km}(x)$ denote the assembly time of a board of type k on machine m induced by some feeder allocation x ($k=1,\ldots,K$ and $m=1,\ldots,M$). With X denoting the set of feasible feeder allocations, the objective function may be specified as follows (compare with (4)):

$$\min_{x\in X}\sum_{k=1}^{K} d_k \max_{m=1,\ldots,M} t_{km}(x). \tag{10}$$

Observe that setup times do not appear in (10) under the assumption of family setups.

In order to write a more complete formulation, let (similarly to the previous section) $j=1,\ldots,J$ denote the feeders to be used, let p_{jkm} be the estimated placement time by machine m of all components of type j on a board of type k, and let C_m be the carrier capacity on machine m. The 0–1 decision variables are

$$x_{jm}=1 \quad \text{if feeder } j \text{ is set up on machine } m$$

for $j=1,\ldots,J$, $m=1,\ldots,M$, and a model for SP2 can be written as

$$\text{minimize} \quad \sum_{k=1}^{K} d_k \max_{m=1,\ldots,M} \sum_{j=1}^{J} p_{jkm}x_{jm}, \tag{11}$$

$$\text{s.t.} \quad \sum_{m=1}^{M} x_{jm} = 1 \quad \text{for all } j, \tag{12}$$

$$\sum_{j=1}^{J} x_{jm} \leqslant C_m \quad \text{for all } m, \tag{13}$$

$$x_{jm}\in\{0,1\} \quad \text{for all } j,m. \tag{14}$$

This model can be linearized by substituting new variables t_k for the max-operators in the objective function (11). This leads to

$$\text{minimize} \quad \sum_{k=1}^{K} d_k t_k \tag{15}$$

$$\text{s.t.} \quad \sum_{m=1}^{M} x_{jm} = 1 \quad \text{for all } j, \tag{16}$$

$$\sum_{j=1}^{J} x_{jm} \leqslant C_m \quad \text{for all } m, \tag{17}$$

$$\sum_{j=1}^{J} p_{jkm}x_{jm} \leqslant t_k \quad \text{for all } k,m, \tag{18}$$

$$x_{jm}\in\{0,1\} \quad \text{for all } j,m. \tag{19}$$

The assembly times p_{jkm} must be roughly estimated, since feeder allocation, feeder location and placement sequencing decisions will eventually interfere with each other to determine the exact assembly time of each board.

Ammons et al. [6] consider a slightly more general feeder allocation model than (15)–(19) by allowing for multiple copies of each feeder type and for partial setups. They solve this mixed integer programming model by branch-and-bound. They mention, however, that (15) provides a poor approximation of the actual makespan when multiple board types are involved.

Crama et al. [18] handle the same objective function and simultaneously solve the feeder allocation and location problems (SP2 and SP5) by local search. Using some of the ideas mentioned in Section 5, they can anticipate on the solution of the placement sequencing problem and are able to obtain close estimates of the actual makespan.

Lapierre et al. [36] consider an integer programming model similar to (15)–(19), but which explicitly incorporates feeder location decisions. They use Lagrangian relaxation techniques to solve it.

Lin and Tardif [39] consider the objective function (15) in a stochastic environment characterized by uncertain demand and machine breakdowns. They propose and solve a stochastic mixed-integer programming formulation of the problem.

6.2.3. Feeder allocation and production sequencing for flexible cells (SP2–SP4)

Consider now a flexible workcell which is to be set up for the assembly of a family of board types $1,\dots,K$. Contrary to the case of assembly lines, production can be assumed here to take place in *mixed mode*, with several types of PCBs circulating simultaneously in the cell. The PCB sequencing subproblem SP4 gains therefore more importance and must be taken into account in the formulation of the feeder allocation problem SP2 (here again, we assume for simplicity that each feeder type can be allocated to one machine only and that SP3 vanishes accordingly).

Integer programming models for SP2 have been proposed by several authors. In one of the earliest papers in this vein, Ammons et al. [7] describe a bicriterion model which simultaneously attempts to achieve workload balance and to minimize the number of visits of each board to the machines. The second objective can be viewed as a proxy for material handling utilization and work-in-process, but also aims at reducing the complexity of the subsequent sequencing problem (SP4). Klincewicz and Rajan [33] (see also [48]) formulate a very similar model in which workload balance is incorporated into the constraints rather than in the objective function. In order to state their model, denote the 0–1 decision variables by

$$x_{jm} = 1 \quad \text{if feeder type } j \text{ is set up on machine } m,$$

$$y_{km} = 1 \quad \text{if board type } k \text{ must visit machine } m.$$

Let d_k denote the number of boards of type k, let p_{jm} be the estimated placement time of all components of type j by machine m, let $\delta_{jk} = 1$ (resp. 0) if PCB type k requires (resp. does not require) feeder j and let T_- (resp. T_+) be a lower bound (resp. upper bound) on the total workload of each machine (i.e., on the makespan of the cell), for $j = 1,\ldots,J$, $k = 1,\ldots,K$, $m = 1,\ldots,M$. The model in [33] is

$$\text{minimize} \quad \sum_{k=1}^{K} d_k \sum_{m=1}^{M} y_{km}, \tag{20}$$

$$\text{s.t.} \quad \sum_{m=1}^{M} x_{jm} = 1 \quad \text{for all } j, \tag{21}$$

$$\sum_{j=1}^{J} x_{jm} \leqslant C_m \quad \text{for all } m, \tag{22}$$

$$\delta_{jk} x_{jm} \leqslant y_{km} \quad \text{for all } j,k,m, \tag{23}$$

$$\sum_{j=1}^{J} p_{jm} x_{jm} \geqslant T_- \quad \text{for all } m, \tag{24}$$

$$\sum_{j=1}^{J} p_{jm} x_{jm} \leqslant T_+ \quad \text{for all } m, \tag{25}$$

$$x_{jm} \in \{0,1\} \quad \text{for all } j,m, \tag{26}$$

$$y_{km} \in \{0,1\} \quad \text{for all } k,m. \tag{27}$$

Klincewicz and Rajan [33] solve this model by a GRASP heuristic. Ammons et al. [7] handle their bicriterion formulation by several heuristic procedures (of the bin packing type for workload balance and of the clustering type for the number of visits) which allow them to put more or less emphasis on each criterion. Another variant of SP2 is proposed by Askin et al. [8]: their objective is to allocate feeders so as to minimize the maximum workload across machines and, simultaneously, to form 'homogeneous' groups of PCBs so as to equalize the assembly time of each PCB within a group. They propose ad hoc heuristics based on similarity measures for the solution of this problem.

Let us now turn to the sequencing subproblem (SP4). This question seems to have been addressed by very few authors. Askin et al. [8] note that, in the framework of flexible cells, problem SP4 resembles the classical open shop scheduling model: given the allocation of feeders to machines (SP2), the assembly of each PCB of type k requires a list of operations $(O_{k1},\ldots,O_{kJ_k})$, where O_{ki} denotes the placement of component i by machine m_i (where m_i is the machine holding component i). The problem consists in defining the start time of each operation so as to minimize the assembly makespan. Notice that the processing time of each operation O_{ki} is not completely determined as long as the remaining subproblems SP5–SP8 have not been solved, but it can usually be reasonably approximated.

After having solved the feeder allocation problem as indicated above, Askin et al. [8] construct a production schedule by applying specialized heuristics from the open shop literature. These heuristics make explicit use of the 'homogeneous' groups of PCBs formed in the first phase.

Lofgren et al. [40] assume that the allocation of feeders to machines (SP2) is given. They focus on a single board type but consider a situation where precedence constraints exist between the assembly operations to be performed on the boards. They attempt to determine a routing of the boards through the shop so as to minimize the number of visits to machines. They reformulate this problem as a linear ordering problem on a directed graph and they analyze the complexity and worst-case performance of approximation algorithms for this problem. They conjecture that, unless $P = NP$, there does not exist a polynomial time algorithm with finite worst case ratio for their model.

Ahmadi and Wurgaft [4] also assume that the allocation of feeders to machines is given and allow for precedence relations among operations, but they explicitly consider multiple PCB types. In order to synchronize the flow of products in the assembly cell, they are interested in finding large subsets of PCBs for which the precedence relations form an acyclic digraph. Alternatively, they propose to determine the smallest number of operations to be replicated so as to remove all cycles from the precedence graph (replicating an operation is roughly equivalent to using multiple copies of a same feeder type; thus, this question is related, in its spirit, to subproblems SP2–SP3).

6.3. Partial setups

Let us turn, finally, to partial setup policies. As mentioned earlier, there exist numerous variants of these strategies, among which:

(b1) *decompose and sequence* [42]: for each PCB type, the feeders loaded on the machines are exactly those required by the bill-of-materials of this board type; between each pair of successive board types, only those changes are performed which are strictly needed;

(b2) some feeders remain permanently on the machines, the other ones are changed as required by the next PCB type to be produced; the decision as to which feeders are permanent or temporary is explicitly incorporated in the optimization process [6,9];

(b3) some feeders are permanently assigned to the machines for reasons which are exogenous to the optimization models [34];

(b4) *partition and repeat* [15,42]: a new feeder setup is performed after all board types have been *partially* processed by the machines; incomplete boards accumulate as work-in-process.

Being intermediate between tear-down and family setups, partial setups clearly provide the most flexibility and allow, in principle, for optimal reduction of the production makespan. The efficiency tradeoff between family setups and (various types of) partial setups has been discussed, for instance, by Ammons et al. [6], Günther et al. [30], Jain et al. [32], Leon and Peters [38], Maimon et al. [41]. More research is needed on this topic (as already mentioned by McGinnis et al. [42]).

When partial setup is used, all subproblems SP1–SP8 become tightly interconnected. In particular, the sequence in which the different types of boards are produced determines the feeders to be loaded and unloaded when a new setup is performed and thus, largely determines the setup time. So, it becomes even more difficult to decouple product grouping, feeder allocation and board sequencing than in the case of family setups: an 'optimal' assignment of products to machine groups is one for which there exists a sequence of board types entailing few feeder changeovers.

These remarks explain that, under the assumption of partial setups, several researchers have linked problems SP1–SP4 to tool switching models investigated in the FMS literature. For a single machine, a well-known tool switching model can be stated as follows: given a family of boards $k=1,\dots,K$, their respective bills-of-materials (described by the parameters δ_{jk}, as in Section 6.2.1) and the feeder carrier capacity C, determine the sequence of boards and the corresponding allocation of feeder types to be loaded on the machines so as to minimize the total number of feeder changeovers. This model, which has close links to the *decompose and sequence* policy, was introduced in a seminal paper by Tang and Denardo [54] (see [16] for a review of the literature on this model). Its connection with PCB assembly was observed by Bard [11]. Jain et al. [32] relied explicitly on this model for a case study on setup optimization at Hewlett–Packard.

When several machines are available for assembly, however, the overall objective of makespan minimization, including setup time *and* assembly time, must be taken into account (since assembly time is influenced by feeder allocation decisions). This objective is not adequately reflected by tool switching models. Therefore, there arises a need for more general models. Such models are proposed by Balakrishnan and Vanderbeck [9] or Ammons et al. [6].

Balakrishnan and Vanderbeck [9] describe a model for product assignment SP1. They postulate that component types are to be partitioned into two classes: *permanent* and *temporary*. Temporary feeders are loaded on the machines as needed and unloaded whenever a batch is completed. With the same generic notations as in Section 6.2.1, let

$y_{ik}=1$ if board type k is assigned to machine group i,

$z_{ij}=1$ if feeder j is set up permanently on machine group i,

$v_{ijk}=1$ if feeder j is set up temporarily on machine group i to assemble board type k,

let b_k be the number of batches of type k to be produced over the planning horizon and let T be an upper-bound on the workload of each machine group ($i=1,\dots,I,\ k=1,\dots,K,\ j=1,\dots,J$). The model is

$$\text{minimize} \quad \sum_{k=1}^{K} b_k \sum_{i=1}^{I}\sum_{j=1}^{J} \delta_{jk} v_{ijk}, \tag{28}$$

$$\text{s.t.} \quad \sum_{i=1}^{I} y_{ik} = 1 \quad \text{for all } k, \tag{29}$$

$$\sum_{j=1}^{J} z_{ij} + \sum_{j=1}^{J} \delta_{jk} v_{ijk} \leqslant N_i \quad \text{for all } i,k, \tag{30}$$

$$\delta_{jk} y_{ik} \leqslant z_{ij} + v_{ijk} \quad \text{for all } i,j,k, \tag{31}$$

$$\sum_{k=1}^{K} a_{ik} y_{ik} \leqslant T \quad \text{for all } i, \tag{32}$$

$$y_{ik} \in \{0,1\} \quad \text{for all } i,k, \tag{33}$$

$$z_{ij} \in \{0,1\} \quad \text{for all } i,j, \tag{34}$$

$$v_{ijk} \in \{0,1\} \quad \text{for all } i,j,k. \tag{35}$$

Balakrishnan and Vanderbeck [9] use a column generation approach to solve this model.

The assumption underlying objective function (28) is that all temporary feeders are removed when assembly of the corresponding batch is completed, even if the same feeder is required by the next board type. This is in contrast with the FMS tool switching model mentioned above, where feeders are assumed to remain on the machine if they are common to successive board types. Removing all feeders, however, allows to reoptimize their location between the production of successive batches (subproblem SP5; FMS tool switching models do not take the location of feeders into account.)

Model (28)–(35) can be viewed as a generalization of (4)–(9). Indeed, ruling out partial setups amounts to setting all variables v_{ijk} to zero. Then, searching for the minimum feasible workload T in (29)–(35) is equivalent to solving the family setup model (4)–(9). On the other hand, when T is very large, model (28)–(35) places the emphasis on setup minimization.

A related model is proposed by Ammons et al. [6] for the allocation of feeders to machines on an assembly line (SP2). These authors develop fast heuristics or use an LP-based branch-and-bound code (MINTO) for the solution of their model.

The above-mentioned models use a rough approximation of the assembly time per board, denoted a_{ik}. Leon and Peters [38] use instead an iterative procedure to obtain more accurate estimates.

7. Conclusions

In this paper, we have reviewed some of the literature on process planning for the optimization of PCB assembly. In our view, some of the most noticeable recent trends in this field have been:

- the consideration of multiple board types in the solution of feeder location and placement sequencing models;
- the development of integer programming models and algorithms for product grouping and feeder allocation subproblems.

More research along these two lines is still needed. In particular, there seems to be a lack of techniques to determine the *global* quality of various solution methods. Indeed, in most practical situations, one needs to resort to heuristic (as opposed to exact) methods to deal with the size and complexity of the optimization problems that arise as part of the planning hierarchy SP1–SP8 described in Section 3. However, few methods are able to give an indication of the global quality of the heuristic solutions produced, of the adequacy of different models, or, for that matter, of the adequacy of the hierarchical decomposition itself.

Finally, one of the aims of this survey has been to facilitate the classification of problems and models found in the literature on PCB assembly. Unfortunately, access to this literature is oftentimes obscured by the fact that the description of the production environment involved and of the problems tackled is insufficiently clear. In order to help readers find their way in forthcoming research publications, we would like to advocate that all authors mention (at least) the following typology elements in their papers:

- shop layout (decoupled workcells, one assembly line, several assembly lines, etc.);
- characteristics of the product mix (high volume—low variety, low volume—high variety, etc.);
- setup policy (see Section 6);
- relevant characteristics of the placement machines (sequential, concurrent, etc.);
- decisions to be taken, according to the list SP1–SP8.

We believe that providing such information would improve communication between research teams and would foster new developments in the field.

Acknowledgements

The authors are grateful to Endre Boros and Peter Hammer for giving them the opportunity and motivation to assemble this survey. They acknowledge the partial financial support of ONR (Grant N00014-92-J-1375), of NATO (Grant CRG931531) and of the European Network DONET (contract number ERB TMRX-CT98-0202).

References

[1] R.H. Ahmadi, A hierarchical approach to design, planning, and control problems in electronic circuit card manufacturing, in: R.K. Sarin (Ed.), Perspectives in Operations Management, Kluwer Academic Publishers, Dordrecht, 1993, pp. 409–429.

[2] J. Ahmadi, R. Ahmadi, H. Matsuo, D. Tirupati, Component fixture partitioning/sequencing for printed circuit board assembly with concurrent operations, Oper. Res. 43 (1995) 444–457.

[3] J. Ahmadi, S. Grotzinger, D. Johnson, Component allocation and partitioning for a dual delivery placement machine, Oper. Res. 36 (1988) 176–191.

[4] R.H. Ahmadi, H. Wurgaft, Design for synchronized flow manufacturing, Manage. Sci. 40 (1994) 1469–1483.

[5] K. Altinkemer, B. Kazaz, M. Köksalan, H. Moskowitz, Optimization of printed circuit board manufacturing: integrated modeling and algorithms, European J. Oper. Res. 124 (2000) 409–421.

[6] J.C. Ammons, M. Carlyle, L. Cranmer, G.W. DePuy, K.P. Ellis, L.F. McGinnis, C.A. Tovey, H. Xu, Component allocation to balance workload in printed circuit card assembly systems, IIE Trans. 29 (1997) 265–275.

[7] J.C. Ammons, C.B. Lofgren, L.F. McGinnis, A large scale machine loading problem in flexible assembly, Ann. Oper. Res. 3 (1985) 319–332.

[8] R.G. Askin, M. Dror, A.J. Vakharia, Printed circuit board family grouping and component allocation for a multimachine, open shop assembly cell, Nav. Res. Logist. 41 (1994) 587–608.

[9] A. Balakrishnan, F. Vanderbeck, A tactical planning model for mixed-model electronics assembly operations, Oper. Res. 47 (3) (1999) 395–409.

[10] M.O. Ball, M.J. Magazine, Sequencing of insertions in printed circuit board assembly, Oper. Res. 36 (1988) 192–201.
[11] J.F. Bard, A heuristic for minimizing the number of tool switches on a flexible machine, IIE Trans. 20 (1988) 382–391.
[12] J.F. Bard, R.W. Clayton, T.A. Feo, Machine setup and component placement in printed circuit board assembly, Int. J. Flexible Manuf. Systems 6 (1994) 5–31.
[13] K. Broad, A. Mason, M. Rönnqvist, M. Frater, Optimal robotic component placement, J. Oper. Res. Soc. 47 (1996) 1343–1354.
[14] J.L. Burbidge, The Introduction of Group Technology, Wiley, New York, 1975.
[15] T.F. Carmon, O.Z. Maimon, E.M. Dar-El, Group set-up for printed circuit board assembly, Int. J. Production Res. 27 (1989) 1795–1810.
[16] Y. Crama, Combinatorial optimization models for production scheduling in automated manufacturing systems, European J. Oper. Res. 99 (1997) 136–153.
[17] Y. Crama, O.E. Flippo, J.J. van de Klundert, F.C.R. Spieksma, The component retrieval problem in printed circuit board assembly, Int. J. Flexible Manuf. Systems 8 (1996) 287–312.
[18] Y. Crama, O.E. Flippo, J.J. van de Klundert, F.C.R. Spieksma, The assembly of printed circuit boards: a case with multiple machines and multiple board types, European J. Oper. Res. 98 (1997) 457–472.
[19] Y. Crama, A.W.J. Kolen, A.G. Oerlemans, F.C.R. Spieksma, Throughput rate optimization in the automated assembly of printed circuit boards, Ann. Oper. Res. 26 (1990) 455–480.
[20] Y. Crama, A.G. Oerlemans, F.C.R. Spieksma, Production Planning in Automated Manufacturing, Springer, Berlin, 1996.
[21] A. Dikos, P.C. Nelson, T.M. Tirpak, W. Wang, Optimization of high-mix printed circuit card assembly using genetic algorithms, Ann. Oper. Res. 75 (1997) 303–324.
[22] Z. Drezner, S. Nof, On optimizing bin picking and insertion plans for assembly robots, IIE Trans. 16 (1984) 262–270.
[23] P.J. Egbelu, C.-T. Wu, R. Pilgaonkar, Robotic assembly of printed circuit boards with component feeder location considerations, Production Planning Control 7 (2) (1996) 162–175.
[24] T.A. Feo, J.F. Bard, S.D. Holland, Facility-wide planning and scheduling of printed wiring board assembly, Oper. Res. 43 (1995) 219–230.
[25] L.R. Foulds, H.W. Hamacher, Optimal bin location and sequencing in printed circuit board assembly, European J. Oper. Res. 66 (1993) 279–290.
[26] R.L. Francis, H.W. Hamacher, C.-Y. Lee, S. Yeralan, Finding placement sequences and bin locations for cartesian robots, IIE Trans. 26 (1994) 47–59.
[27] B. Gaboune, G. Laporte, F. Soumis, Optimal strip sequencing strategies for flexible manufacturing operations in two and three dimensions, Int. J. Flexible Manuf. Systems 6 (1994) 123–135.
[28] A.E. Gray, A. Seidmann, K.E. Stecke, A synthesis of decision models for tool management in automated manufacturing, Manage. Sci. 39 (1993) 549–567.
[29] S. Grotzinger, Feeder assignment models for concurrent placement machines, IIE Trans. 24 (1992) 31–46.
[30] H.O. Günther, M. Gronalt, R. Zeller, Job sequencing and component set-up on a surface mount placement machine, Production Planning Control 9 (1998) 201–211.
[31] M.S. Hillier, M.L. Brandeau, Optimal component assignment and board grouping in printed circuit board assembly, Oper. Res. 46 (1998) 675–689.
[32] S. Jain, M.E. Johnson, F. Safai, Implementing setup optimization on the shop floor, Oper. Res. 44 (1996) 843–851.
[33] J.G. Klincewicz, A. Rajan, Using GRASP to solve the component grouping problem, Naval Res. Logist. 41 (1994) 893–912.
[34] C. Klomp, J. van de Klundert, F.C.R. Spieksma, S. Voogt, The feeder rack assignment problem in PCB assembly: a case-study, Int. J. Prod. Econom. 64 (2000) 399–407.
[35] P.J.M. van Laarhoven, W.H.M. Zijm, Production preparation and numerical control in PCB assembly, Int. J. Flexible Manuf. Systems 5 (1993) 187–207.
[36] S.D. Lapierre, L. De Bargis, F. Soumis, Balancing printed circuit board assembly line systems, Int. J. Production Res. 38 (2000) 3899–3911.

[37] T. Leipälä, O. Nevalainen, Optimization of the movements of a component placement machine, European J. Oper. Res. 38 (1989) 167–177.
[38] V.J. Leon, B.A. Peters, Replanning and analysis of partial setup strategies in printed circuit board assembly systems, Int. J. Flexible Manuf. Systems 8 (1996) 389–412.
[39] W.-L. Lin, V. Tardif, Component partitioning under demand and capacity uncertainty in printed circuit board assembly, Int. J. Flexible Manuf. Systems 11 (1999) 159–176.
[40] C.B. Lofgren, L.F. McGinnis, C.A. Tovey, Routing printed circuit cards through an assembly cell, Oper. Res. 39 (1991) 992–1004.
[41] O.Z. Maimon, E.M. Dar-El, T.F. Carmon, Set-up saving schemes for printed circuit boards assembly, European J. Oper. Res. 70 (1993) 177–190.
[42] L.F. McGinnis, J.C. Ammons, M. Carlyle, L. Cranmer, G.W. DePuy, K.P. Ellis, C.A. Tovey, H. Xu, Automated process planning for printed circuit card assembly, IIE Trans. 24 (1992) 18–30.
[43] A. Mody, R. Suri, M. Tatikonda, Keeping pace with change: international competition in printed circuit board assembly, Ind. Corporate Change 4 (1995) 583–613.
[44] L.K. Moyer, S.M. Gupta, SMT feeder slot assignment for predetermined component placement paths, J. Electron. Manuf. 6 (1996) 173–192.
[45] L.K. Moyer, S.M. Gupta, Simultaneous component sequencing and feeder assignment for high speed chip shooter machines, J. Electron. Manuf. 6 (1996) 271–305.
[46] O.Z. Maimon, A. Shtub, Role of similarly measures in PCB grouping procedures, Int. J. Production Res. 30 (1992) 973–983.
[47] H. Nakahara, PCB output 1998, Printed Circuit Fabrication, June 1999.
[48] A. Rajan, M. Segal, Assigning components to robotic workcells for electronic assembly, AT & T Technical J. 68 (1989) 93–102.
[49] M.S. Sodhi, A. Agnetis, R.G. Askin, Tool addition strategies for flexible manufacturing systems, Int. J. Flexible Manuf. Systems 6 (1994) 287–310.
[50] K.E. Stecke, Formulation and solution of nonlinear integer production planning problems for flexible manufacturing systems, Manage. Sci. 29 (1983) 273–288.
[51] K.E. Stecke, I. Kim, A study of part type selection approaches for short-term production planning, Int. J. Flexible Manuf. Systems 1 (1988) 7–29.
[52] Y.S. Su, C. Wang, P.J. Egbelu, D.J. Cannon, A dynamic points specification approach to sequencing robot moves for PCB assembly, Int. J. Comput. Integr. Manuf. 8 (1995) 448–456.
[53] F.F. Suarez, M.A. Cusumano, C.H. Fine, An empirical study of manufacturing flexibility in printed circuit board assembly, Oper. Res. 44 (1996) 223–240.
[54] C.S. Tang, E.V. Denardo, Models arising from a flexible manufacturing machine, Part I: Minimization of the number of tool switches, Oper. Res. 36 (1988) 767–777.
[55] C.S. Tang, E.V. Denardo, Models arising from a flexible manufacturing machine, Part II: Minimization of the number of switching instants, Oper. Res. 36 (1988) 778–784.
[56] J.L. van Zante-de Fokkert, T.G. de Kok, The simultaneous determination of the assignment of items to resources, the cycle times, and the reorder intervals in repetitive PCB assembly, Ann. Oper. Res. 92 (1999) 381–401.
[57] I. Viczián, Finding placement sequences and bin locations for cartesian robots, A working paper of the University of Würzburg, 1993.
[58] R.A. Walas, R.G. Askin, An algorithm for NC turret punch press tool location and hit sequencing, IIE Trans. 16 (1984) 280–287.
[59] C. Wang, L. Ho, D.J. Cannon, Heuristics for assembly sequencing and relative magazine assignment for robotic assembly, Comput. Ind. Eng. 34 (1998) 423–431.

For further reading

A. Agnetis, R.G. Askin, M.S. Sodhi, Tool addition strategies for flexible manufacturing systems, Int. J. Flexible Manuf. Systems 6 (1994) 287–310.

R.H. Ahmadi, P. Kouvelis, Staging problem of a dual delivery pick-and-place machine in printed circuit card assembly, Oper. Res. 42 (1994) 81–91.

R. Tyler, View from Europe, Printed Circuit Fabrication, February 1999.
S. Voogt, Short term scheduling in PCB assembly, Philips Report CTR 597-93-0106, 1993.
T.A. Younis, T.M. Cavalier, On locating part bins in a constrained layout area for an automated assembly process, Comput. Ind. Eng. 18 (1990) 111–118.