A Decomposition Approach with Inserted Idle Time Scheduling Subproblems in Group Scheduling

Cumhur A. Gelogullari[1] and Rasaratnam Logendran[2]

[1] Operations Research and Decision Support, American Airlines,
4333 Amon Carter Blvd MD 5358 Fort Worth, TX 76155, U.S.A.
cumhur.gelogullari@aa.com

[2] Department of Industrial and Manufacturing Engineering, 118 Covell Hall,
Oregon State University, Corvallis, OR 97331, U.S.A.
logendrr@engr.orst.edu

Abstract. This paper focuses on minimizing the makespan of a multi-machine flowshop group-scheduling problem that is typically found in the assembly of printed circuit boards, which is characterized as one with carryover sequence-dependent setup times. The intent is to minimize the makespan of schedules comprised of the sequence of board groups as well as the sequence of board types within each group. Specifically, the models and algorithms developed for identifying strong lower bounds on the optimal/near optimal solutions within a reasonable computation time are emphasized. The efficacy of the lower bound developed is demonstrated by using it to quantify the quality of a heuristic solution for the same problem, developed based on tabu search. To obtain strong lower bounds, the problem is decomposed into a master problem and single-machine subproblems which, except for the subproblem on the first machine, are inserted idle time scheduling problems. A tabu search based heuristic is developed to solve the subproblems approximately. Each solution found during the tabu search process is evaluated using a timetabling problem that is formulated as a simple integer program for identifying the inserted optimal idle times on the machine in order to minimize the subproblem objective function. The column generation algorithm developed for the decomposed problem is demonstrated on a real problem obtained from the industry.

Keywords: Carryover sequence dependent setup times, printed circuit board assembly, mathematical programming, tabu search, column generation

1 Introduction, Motivation, and Problem Summary

This paper addresses the multi-machine flowshop group-scheduling problem with carryover sequence-dependent setup times for minimizing the makespan. The problem is typically found in the assembly of printed circuit boards (PCBs) in electronics manufacturing. In order to solve the problem efficiently and effectively, high-level metasearch heuristics based on tabu search have been developed [3]. The

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 412–415. ISBN 80-210-3726-1.

primary focus of this paper is to identify strong lower bounds on the optimal makespan within a reasonable computation time, so that the quality of a heuristic solution, i.e. an upper bound, can be quantified as its percentage deviation from the lower bound. A mathematical programming decomposition approach is developed to obtain strong lower bounds, which, interestingly, involves *timetabling problems* as part of an efficient approximation algorithm used to solve the subproblems in a column generation algorithm.

A PCB is a laminated board assembled with a dozen to thousands of electronic components. The PCB assembly is performed on *automated placement machines* that insert the components on the boards quickly and reliably. Before starting production of a PCB on a machine, the required components are loaded on the appropriate feeders during a setup operation. It is not practical to keep changing the component feeders on a frequent basis for each individual board. Typically, in electronics manufacturing, different board types requiring similar components are grouped together and a single setup operation is performed for each board group. The intent is to load all of the components required of the board types in a group on the proper feeders in *one setup*, just so that the board types in that group can be produced one after the other. As a result, the scheduling problem considered here falls under the category known as *group scheduling* and the scheduling decisions must be addressed at two levels. The problem at the first level is associated with the individual board types within groups and is referred to as the "board level" problem. For the board level problem, a sequence of boards in each group must be determined to minimize the makespan, while different board groups themselves must be sequenced at the "group level" so as to minimize the same performance measure.

The challenges encountered in group scheduling in PCB manufacturing are far greater and distinctly different from that in traditional hardware manufacturing. Although the placement machines automate the PCB assembly processes, high-speed and precise operation and especially the flexibility in tooling makes it a difficult task to control the operations on them. The setup time required of a group of PCBs on a machine is dependent on the configuration of the components on the machines, which in turn depends on *not just on the immediately preceding group*, but on *all of the preceding groups* and the *order* in which they were processed. In a sense, the setup times are not only sequence-dependent, but also *carried over* from the very first board group to the one currently being considered for production. This makes the relationships among setup times of board groups highly complicated compared to the sequence-independent or sequence-dependent setup times encountered in traditional hardware manufacturing. In other words, they can be defined easily in the latter, while the carryover sequence-dependent setup times are hard to explicitly define. In addition, the PCB assembly processes considered here are performed on multiple sequential machines.

2 A Mathematical Programming Decomposition Approach and Timetabling

This paper presents a novel mixed-integer linear programming (LP) formulation of the problem. This problem was addressed by a few researchers previously. However, all of the previous research simplifies the problem either by approximating the setup times as sequence-independent or sequence-dependent setup times, which results in loosing valuable information and identifying inferior solutions. Our model, on the other hand, considers the setup times explicitly, hence introduces a greater degree of accuracy into the problem formulation than before. To obtain strong lower bounds, the problem is decomposed into a master problem and single-machine subproblems which, except for the subproblem on the first machine, are *inserted idle time* scheduling problems [4]. Typically, in an inserted idle time scheduling problem, the objective function is not a regular function and the machine can be kept idle for some time when it could begin processing an operation.

We essentially reformulate the problem as an integer programming problem with an exponential number of variables, each representing a *schedule*. A column generation (CG) algorithm is developed to solve the LP relaxation of the master problem. When solving the subproblems in the CG algorithm, a two-phase approach is employed. In the first phase, the subproblems are solved approximately with a fast tabu search algorithm - tabu search column generator (TSCG) - as long as TSCG identifies new columns. Starting with an initial sequence of board groups and board types within each group, TSCG keeps generating new sequences by performing simple exchange moves. Each sequence (or move) must be evaluated. However, since the subproblems are inserted idle time scheduling problems, it is not straight forward how to evaluate the objective function value for each sequence identified during the search. In this context, therefore, each time a sequence is to be evaluated, we need to *timetable* the board types to insert *optimal* idle times on the machine in order to minimize the subproblem objective function. Such a timetabling problem is formulated as a simple integer program.

It is well known that CG algorithms are prone to the so-called tailing-off effect [1]. While usually an optimal solution is approached considerably fast, it may take a very long time to prove LP optimality. Recently, several methods have been suggested against this drawback. Our approach to stabilize and accelerate the CG algorithm is similar to the one proposed by du Merle et al. [2]. First, a bound is imposed on the dual variables by introducing artificial variables into the LP master problem (LMP) and new negative reduced cost columns are generated until the subproblems fail to do so. If some of the artificial variable values are nonzero, the bound imposed on the dual variables is slightly relaxed at that instance, and the process similar to the one before is continued until all the artificial variable values are zero and no negative reduced column can be identified. In our case, the benefit of introducing artificial variables is two fold: First, the dual variables are initially constrained in some interval so that they smoothly converge to their optimal values. Second, we constrain the dual variables in such a way that the subproblem objective functions - except for the subproblem corresponding to the last machine - become regular and board types can be timetabled without any idle time on the machines. Furthermore, it is shown that

when dual variables are bounded in the subproblems, board types within each group follow the shortest run time rule, which makes the subproblems much easier to solve.

The optimal solution to LMP usually provides a strong lower bound on the optimal makespan, but it is not necessarily integral. Better lower bounds and integral solutions can be identified by branching, resulting in a branch-and-price (B&P) algorithm. The paper develops efficient branching rules that are compatible with the CG algorithm. The idea is based on avoiding generation of sequences in which certain board types are assigned to certain positions on one branch and also allowing them on the other branch.

3 Conclusion

The CG algorithm is demonstrated on a real problem obtained from the industry. The lower bound on the makespan identified by the CG algorithm after only a few iterations is within 2% of an upper bound, and the lower bound obtained from the LP relaxation of the original formulation is shown to be highly inferior to the CG lower bound.

References

1. Barnhart, C., Johnson E., Nemhauser, G., Savelsbergh, M., Vance, P.: Branch-and-Price: Column Generation for Solving Huge Integer Programs. *Oper. Res.* 46 (1998) 316-329
2. du Merle, O., Villeneuve, D., Desrosiers, J., Hansen, P.: Stabilized Column Generation. *Discrete Math.* 194 (1999) 229-237
3. Gelogullari, C.A.: Group Scheduling Problems in Electronics Manufacturing. Doctoral Dissertation, Oregon State University, Corvallis, OR, USA (2005).
4. Kanet, J.J., Sridharan, V.: Scheduling with Inserted Idle Time: Problem Taxonomy and Literature Review. *Oper. Res.* 48(2000) 99-110