European Journal of Operational Research 217 (2012) 259–269

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

On a learning precedence graph concept for the automotive industry

Hanne Klindworth, Christian Otto, Armin Scholl *

Friedrich-Schiller-University of Jena, Chair of Management Science, Carl-Zeiß-Straße 3, D-07743 Jena, Germany

ARTICLE INFO

Article history:
Received 30 March 2011
Accepted 11 September 2011
Available online 29 September 2011

Keywords:
Assembly line balancing
Precedence graph
Learning approach
Production process
Decision support

ABSTRACT

Assembly line balancing problems (ALBP) consist in assigning the total workload for manufacturing a product to stations of an assembly line as typically applied in automotive industry. The assignment of tasks to stations is due to restrictions which can be expressed in a precedence graph. However, (automotive) manufacturers usually do not have sufficient information on their precedence graphs. As a consequence, the elaborate solution procedures for different versions of ALBP developed by more than 50 years of intensive research are often not applicable in practice.

Unfortunately, the known approaches for precedence graph generation are not suitable for the conditions in the automotive industry. Therefore, we describe a new graph generation approach that is based on learning from past feasible production sequences and forms a sufficient precedence graph that guarantees feasible line balances. Computational experiments indicate that the proposed procedure is able to approximate the real precedence graph sufficiently well to detect optimal or nearly optimal solutions for a well-known benchmark data set. Even for additional large instances with up to 1,000 tasks, considerable improvements of line balances are possible. Thus, the new approach seems to be a major step to close the gap between theoretical line balancing research and practice of assembly line planning.

1. Introduction

Many modern production systems, as, e.g., in automotive industry, rely on the principle of assembly line work (cf. Boysen et al., 2008). The units of the products to be assembled move down a (paced) assembly line, which is composed of successive stations coupled by a conveyor belt or a similar transportation system. In each station, one or more workplaces are installed. Usually, each workplace is equipped with one worker who performs a set of tasks on each of the successive product units in a cyclical manner observing the cycle time available for each workpiece (on average). Tasks are elementary operations, obtained by breaking down the total work content into basic motions with deterministic times. For this purpose predetermined motion time systems like MTM (Methods-Time Measurement) or MOST (Maynard Operation Sequence Technique) are utilized (cf. Maynard et al., 1948; Karger and Bayha, 1987; Kanawaty, 1992; Longo and Mirabelli, 2009). The order of processing the tasks is restricted by technological and organizational conditions, e.g., mounting a radio device requires having installed a fixture and cables before. These partial orderings of tasks are collected within a precedence graph.

When such an assembly line production system has to be installed or modified, e.g., due to changes in the production process or the demand structure, the assembly line balancing problem (ALBP) has to be solved. It is to assign the tasks to the stations (workplaces) based on, among others, the precedence graph (for details, see Section 2).

In the automotive industry, a typical information and planning system contains, among other data not relevant in this context, the description of tasks including their deterministic task times, e.g., derived by some MTM approach, the current assignment of tasks to workplaces (and, thus, stations) and the execution sequences of tasks within each workplace. However, almost no precedence relations are documented, not to mention an entire precedence graph. The huge manual input and the multitude of tasks (up to several hundreds or even thousands) prevent manufacturers from collecting and maintaining precedence relations (cf. Ammer, 1985, p. 17). This absence of documented information on precedence relations is the main obstacle in applying the arsenal of well explored theoretical assembly line balancing methods in practice.

In practice, planning, balancing and controlling assembly lines are based on subdividing the production processes and, hence, the assembly lines into segments. Each segment is managed by a dedicated human planner, who becomes an expert for this part of the system. Though some software systems provide a component for automatic line balancing, the planners mostly balance their segments of the line by manually shifting tasks from one station to another, because precedence data is not available or existent data is not reliable. This is a very time-consuming and fault-prone job, which is solely driven by the experience and

* Corresponding author. Tel.: +49 3641 943171; fax: +49 6913303398436.
E-mail addresses: armin.scholl@uni-jena.de, a.scholl@wiwi.uni-jena.de (A. Scholl).

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.09.024

knowledge of planners. By appending the plans of succeeding line segments, the entire production plan is developed. It represents the best of all the knowledge and experience of all contributing planners, including a lot of implicit precedence information, which we will exploit in our approach.

Preliminary case studies, we performed at several real-world lines, show that there is a great potential for improving balances, accelerating planning and making the planning process more independent of particular planners. Also, different major German car manufacturers, with whom we co-operate, desire to have precedence graphs on hand as they are aware of the disadvantages of balancing lines without this important piece of information. In order to close this gap, we propose an approach to generate precedence graphs (almost) automatically by learning precedence constraints from prior feasible task sequences, which are, as a rule, very well documented by manufacturers.

The paper is organized as follows: Section 2 considers selected terms and aspects of assembly line balancing which are required for understanding and examining of the proposed approach. In Section 3, we review former approaches for determining or generating precedence information described in the academic literature and/or applied in related areas. The new concept of learning a precedence graph from past production sequences is presented in Section 4. Section 5 reports on computational experiments which show the usefulness of the new approach. Summary and conclusions are given in Section 6.

2. Basics of assembly line balancing

Each assembly line can be modeled as a sequence of m stations $k = 1, \ldots, m$. In order to facilitate presentation, we assume that each station contains a single workplace equipped with one worker. The production process is subdivided into n tasks which are collected in (node) set $V = \{1,\ldots,n\}$. Execution of task $j \in V$ takes task time t_j. At each station k, a specific set of tasks, called station load S_k, is repeatedly executed. The station time $t(S_k) = \sum_{j \in S_k} t_j$ must not exceed a given cycle time c available per workcycle.

The evolving decision problem of assigning tasks to stations in order to optimize a given objective function is known as the assembly line balancing problem (ALBP). Possible objectives include cost minimization, profit maximization, and maximization of line efficiency (e.g., minimization of the number of stations given the cycle time). For a detailed problem statement as well as surveys and classification schemes for ALBP, we refer to Baybars (1986), Becker and Scholl (2006), and Boysen et al. (2007, 2008).

Precedence constraints can be summarized and visualized in a precedence graph. The precedence graph is an acyclic digraph $G = (V,E)$ that contains a node for each task $j \in V$ and an arc $(i,j) \in E$ for each non-redundant precedence relation which requires that task $i \in V$ is finished before another task $j \in V$ can be started. The task times t_j are allocated as node weights.

Fig. 1 shows an example of a precedence graph with six tasks and task times of 1 or 4 time units. Task 1 has tasks 2 and 3 as direct successors as well as tasks 4, 5 and 6 as indirect successors, i.e., task 1 has to be completed before any other task can be started. For task 6 to be executed, task 1 (indirect predecessor) and task 3 (direct predecessor) have to be finished. Besides precedence constraints, the precedence graph also shows (precedence) independencies of tasks. Tasks 4 and 6 exemplify this issue as they can be executed in either order. Two tasks i and j which are not related by precedence are called *independent* of each other.

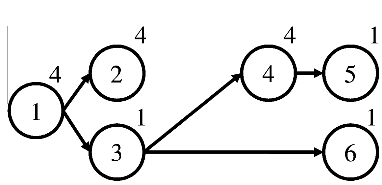

Fig. 1. Example Precedence Graph.

Traditionally in the literature, each feasible solution of an ALBP instance is defined by station loads $S_k \subset V$ of the stations $k = 1, \ldots, m$ which fulfill the given constraints. In practice, however, production plans also contain information about practicable sequences of tasks π^k in each station k. So we can build a complete sequence π, which represents at least one feasible ALBP solution, by appending the station specific sequences π^k consecutively.

We use the well-known simple ALBP with the objective of minimizing the number of stations given the cycle time (SALBP-1) as an example problem throughout the paper (cf. Baybars, 1986; Scholl and Becker, 2006). However, this is not a restriction but facilitates analysis. It is a particular strength of our approach to be simple and general such that it can also be applied to any generalized ALBP as long as the basic structure of precedence constraints is not affected. It has even more relevance, as more degrees of freedom (by getting to know much more precedence-feasible sequences) enable planers to fulfill the other constraints in a better way, i.e., to utilize time and space more efficiently. Even the mixed-model production setting and parallel workplaces as the most typical generalizations in automotive and related industries (cf. Boysen et al., 2008; Becker and Scholl, 2009) can be handled easily. As a standard (in theory and practice), a joint precedence graph is defined in case of different models being produced in intermixed sequence at the same line; for a method based on demand forecasts see Boysen et al. (2009a). Parallel workplaces even increase the degrees of freedom in our approach as production planners pay attention to only assigning tasks which are independent of each other to different workplaces of the same station.

Note that minimizing the number of stations or, more generally, workplaces is still the most relevant objective in automotive industry as this value determines the staff required which is the most important cost driver without being able to estimate cost effects other than wages precisely enough. Of course, the learning precedence graph approach is also applicable to balancing problems with other objectives like minimization of cycle time or maximizing line efficiency.

3. Analyzing previous work on generating precedence information

In the literature, we find some research work related to deriving precedence information. However, most researchers do not aim at constructing precedence graphs but to find feasible sequences.

Such concepts can be categorized in two main classes: on the one hand, there are manual and automated approaches that are intended to detect *all* feasible sequences. On the other hand, genetic algorithms and case-based reasoning procedures are applied to search for *a few good* sequences.

The earliest approach to find feasible assembly sequences was introduced by Bourjault (1984). During his question-based procedure, an expert has to decide on the feasibility of assembly actions. However, as the number of liaisons between parts (connection points that correspond to mounting operations) rises, the number of questions grows exponentially. De Fazio and Whitney (1987) modified Bourjault's approach and reduced the number of questions to two times the number of liaisons. While both methods are based on the assembly of the product, Homen de Mello and Sanderson (1990, 1991) pioneered the disassembly approach. The basic idea is to enumerate all possibilities to disassemble a product