Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 194 (2009) 787–794
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Production, Manufacturing and Logistics

A dynamic programming based heuristic for the assembly line balancing problem

Joaquín Bautista [a], Jordi Pereira [b,*]

[a] Nissan Chair, Universitat Politècnica de Catalunya, Avda. Diagonal 647, 08028 Barcelona, Spain
[b] Escola Tècnica Superior d'Enginyeria Industrial de Barcelona, Departament d'Organizació d'Empreses, Universitat Politècnica de Catalunya, Avda. Diagonal 647, 7th floor, 08028 Barcelona, Spain

Received 26 October 2007; accepted 14 January 2008
Available online 26 January 2008

Abstract

The simple assembly line balancing problem is the simplification of a real problem associated to the assignment of the elementary tasks required for assembly of a product in an assembly line. This problem has been extensively studied in the literature for more than half a century. The present work proposes a new procedure to solve the problem we call Bounded Dynamic Programming. This use of the term Bounded is associated not only with the use of bounds to reduce the state space but also to the reduction of such space based on heuristics. This procedure is capable of obtaining an optimal solution rate of 267 out of 269 instances, which have been used in previous works, thus obtaining the best-known performance for the problem. These results are an improvement from any previous procedure found in the literature even when using smaller computing times.

Keywords: Car manufacturing; Production; Bounded Dynamic Programming; Assembly line balancing

1. Introduction

The assembly line balancing problem (ALBP) is a classic problem that has been tackled over the last 50 years. A simplified view of the problem defines the balancing of an assembly line as the grouping of the tasks required to assemble the final product to the workstations conforming the assembly line. The stations are usually arranged in a serial fashion and are linked together by a transport system whose primary mission consists of moving the product from one workstation to the next. Each station presents a workload equivalent to the tasks assigned to that station. Once the permanent manufacturing conditions have been achieved, the production items flow along the line at a constant rate, and each station has an equal allotted time to finish their tasks. This is known as cycle time.

Usually, the objective of the balancing problem is the minimization of the idle time of the line via the minimization of the number of required stations, the minimization of the cycle time, or a combination of both. It is also possible to define a feasibility problem in the case that both the cycle time and the number of stations are fixed.

A classification proposed by Baybars (1986) divides all balancing problems into two classes: a first class of problems known as simple, SALBP, whose members are clearly stated in the aforementioned work, and a second class of problems known as generals, GALBP, which is constituted of all other problems not belonging to the SALBP class. The SALBP class is constituted of assembly problems where only two kinds of task assignment constraints are taken into account in relation to the stations

* Corresponding author. Tel.: +34 934010769; fax: +34 934016054.
E-mail addresses: joaquin.bautista@upc.edu (J. Bautista), jorge.pereira@upc.edu (J. Pereira).

0377-2217/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2008.01.016

(1) Cumulative constraints associated to the available time of work in the stations.
(2) Precedence constraints created by the requirement of some tasks to be performed after other tasks have been finished.

Any other problem taking into account any additional considerations like incompatibilities between tasks, different line shapes, space constraints or parallel stations, between many others, are included in the GALBP class.

Even if the majority of real assembly lines belong to the GALBP class, the scientific community has invested a great deal of effort to develop efficient procedures to solve the SALBP issues. This is primarily due to the fact that the SALBP model is the underlying model of many GALBP cases and the procedures developed for SALBP can be adapted, with greater or smaller difficulty, to GALBP. Amongst the various SALBP models, differentiated by the objective function they use, the most typically studied model is the SALBP-1. SALBP-1 consists in minimizing the number of required stations for a given cycle time.

The present work studies the resolution of SALBP-1 and puts forward an algorithm based on how Dynamic Programming can solve SALBP-1. The proposed procedure has shown vast improvements in comparison to the results given by all other algorithms found in the literature to date, obtaining 267 optimal solutions out of the 269 instances found in the literature. As well, it has equaled the best-known solution for one of the other two remaining instances.

The rest of the paper is structured as follows: Section 2 studies the previous work found in the literature relative to the present study, while Section 3 is devoted to a formal description of the treated problem and presents the Dynamic Programming formulation used to solve it. Section 4 studies the proposed algorithm and its most important characteristics. Section 5 shows the results obtained by the algorithm from a computational experience with benchmark instances from the literature, and finally Section 6 shows the conclusions of the present work.

2. Literature review

The literature includes a large number of procedures in solving the simple assembly line balancing problem, as seen in a recent study (see Scholl and Becker, 2006) as well as two different studies based on the description and resolution of problems belonging to the General class (see Becker and Scholl, 2006 or Boysen et al., 2007). The present section will be devoted to the proposals of special interest for the present work.

The procedures to solve assembly lines can be categorized in three groups:

(1) A first group composed by constructive or "greedy" procedures. These procedures make use of a static or dynamic priority rule to assign tasks to different workstations (see Talbot et al., 1986). These procedures offer relatively good solutions in very reduced computing times.
(2) A second group is composed of enumeration procedures, usually tree search based procedures like branch and bound or graph based ones like Dynamic Programming (see Scholl, 1999). Of the different procedures of this class, we highlight SALOME (Scholl and Klein, 1997) a Branch and Bound algorithm capable of optimally solving 260 out of the 269 cases from the literature used for comparison purposes.
(3) The third and final group is composed by different metaheuristic approaches. Although, the results are not usually as good as those provided by the second group procedures, they are still an important area of research due to the higher applicability of these procedures to general problems. This is due to the dependency of exact procedures to use good bounds to find good solutions. Amongst the most effective procedures for the problem are the Tabu Search proposals by Scholl and Voß (1996) or Lapierre et al. (2006), the Ant algorithm from Bautista and Pereira (2007) and the Beam-ACO algorithm from Blum et al. (2006). The latter one have proven to be the most successful, being able to optimally solve 245 out of the 269 instances of the literature.

We give three possible reasons to answer the question of why enumeration procedures are better than heuristics ones: (1) the quality of constructive procedures, which are able to solve many instances from the literature without the help of any other mechanism (see Scholl, 1999), (2) the existence of good quality bounding procedures with very limited computing time requirements (see again Scholl, 1999), and (3) the difficulty to define good search neighborhoods for SALBP-1 (see Bautista and Pereira, 2007 for the latter point). As an example of the aforementioned topic, the algorithm of Blum et al. (2006) is a partial enumeration procedure with no local search phase to improve constructed solutions.

To a greater or lesser extent, the lack of results from local search procedures explains the quantity of papers in the field based on tree search approaches. Of the proposed works, we highlight the paper from Fleszar and Hindi (2003) where an improvement of the previous procedure by Hoffmann (1963) is developed.

The procedure proposed by Fleszar and Hindi is based on the enumeration of the possible assignments of tasks to each station, in turn, choosing an assignment for a station that allows for the minimum possible idle time. The procedure can be seen as the sum of different enumeration procedures for each separate station. The procedure makes use of bounds and other known properties of the problem to improve the algorithm previously defined, solving over 200 instances from the benchmark set to optimality with extremely low computing times.