Available online at www.sciencedirect.com

ScienceDirect

European Journal of Operational Research 177 (2007) 2016–2032

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/ejor

Ant algorithms for a time and space constrained assembly line balancing problem

Joaquín Bautista *, Jordi Pereira

Nissan Chair ETSEIB Universitat Politècnica de Catalunya, 7th Floor, Avda Diagonal 647, 08028 Barcelona, Spain

---

Abstract

The present article focuses on the application of a procedure based on ant colonies to solve an assembly line balancing problem. After an introduction to assembly line problems, the problem under study is presented: the Time and Space constrained Assembly Line Balancing Problem (TSALBP); and a basic model of one of its variants is put forward for study. Subsequently, an ant algorithm is presented that incorporates some ideas that have offered good results with simple balancing problems. Finally, the validity of the proposed algorithms is tested by means of a computational experience with reference instances, and the conclusions of the study are presented.

Keywords: Artificial intelligence; Manufacturing; Metaheuristics; Production

---

1. Introduction

An assembly line is made up of a number of workstations, m, arranged in series and in parallel, through which the work in progress on a product flows. The stations are linked together by a transport system whose mission is to supply materials to the main flow and to move the production items from one station to the next. Production items may be of one type (single-model) or of several types (mixed-model).

The manufacturing of a production item is divided up into a set V of n tasks; each station k ($1 \leqslant k \leqslant m$) is assigned a subset of tasks S_k ($S_k \subseteq V$) called the *workload* of station k; a task j may only be assigned to one station. Each task j requires an operation time $t_j > 0$ for its execution that is determined as a function of the manufacturing technologies and the employed resources. In mixed-model assembly lines, it is usual to

---

* Corresponding author. Tel.: +34 93 401 17 03; fax: +34 93 401 60 54.
E-mail addresses: joaquin.bautista@upc.edu (J. Bautista), jorge.pereira@upc.edu (J. Pereira).

doi:10.1016/j.ejor.2005.12.017

calculate operation times weighting the presence of each model in the mix. If the range of required operation times between different models is high, units must be sequenced accordingly.

Additionally, the technology and nature of the product itself mean that each task j has a set of immediately predecessor tasks, P_j, which must be accomplished before commencing task j. These constraints are normally represented by means of an acyclic precedence graph whose vertices represent the tasks and in which each directed arc (i,j) indicates that task i must be finished before commencing task j on the line; thus, if $i \in S_h$ and $j \in S_k$, then $h \leqslant k$ must be fulfilled.

Each station k presents a station workload time $t(S_k)$ that is equal to the sum of the durations of the tasks assigned to station k. Once permanent manufacturing conditions have been achieved, the production items flow along the line at a constant rate, and each station k has a time c, called the cycle time, to carry out its assigned tasks. On automotive trim or body lines, the items normally remain in spaces allotted to each station, where the corresponding tasks are executed during a time equal to the cycle. The items are then transferred to the next station in an insignificant period of time, thus initiating a new cycle.

The cycle time c determines the production rate r of the line ($r = 1/c$), and cannot be less than the maximum station workload time: $c \geqslant \max_k\{t(S_k)\}$, nor must it be greater than the sum of the durations of the tasks of V: $c \leqslant \sum_k t(S_k) = t_{\text{sum}}$. Each station k presents a idle time $I_k = c - t(S_k)$. The sum of these partial times gives rise to the total idle time, $I_{\text{sum}} = \sum_k I_k = m \cdot c - t_{\text{sum}}$, which is related to the inefficiency of the line.

In general, ALBP (Assembly Line Balancing Problems) focus on grouping together the tasks in the set V in workstations in an efficient and coherent way. In short, the goal is to achieve a grouping of tasks that minimizes the inefficiency of the line or its total downtime and that respects all the constraints imposed on the tasks and on the stations. ALBP belongs to the general class of sequencing problems (see Baker, 1974), and can be seen as Bin Packing Problems with additional side constraints, as precedence relationships between tasks. These precedence constraints establish an implicit order of bins, deriving in a sequence of operations.

A first family of problems, known as SALBP (Simple Assembly Line Balancing Problems) Baybars (1986), may be stated in the following way: given a set of n tasks with their attributes and a precedence graph, each task must be assigned to a single station in such a way that all the precedence constraints are satisfied and no station workload time, $t(S_k)$, is greater than the cycle time c.

The SALBP family presents four variants: SALBP-1: minimize the number of stations m given a fixed value of the cycle time c; SALBP-2: minimize the cycle time c (maximize the production rate r) given a fixed number of stations m; SALBP-E: simultaneously minimize c and m considering their relation with the total idle time or the inefficiency of the line; SALBP-F: given m and c, determine the feasibility of the problem, and if it is feasible, find a solution.

When other considerations are added to those of the SALBP family, the problems are known in the literature by the name of GALBP (General Assembly Line Balancing Problems). This family includes those problems with additional constraints, such as for instance the consideration of parallel stations, Daganzo and Blumfield (1994) and Vilarinho and Simaria (2002), forced groupings of tasks, Deckro (1989) possible incompatibilities among tasks, Agnetis et al. (1995) and differences between workstations, Nicosia et al. (2002), among others. An up-to-date analysis of the bibliography and available state of the art procedures can be found in Scholl and Becker (2006) for SALBP family of problems, and Becker and Scholl (2006) for GALBP ones.

As regards problem-solving procedures, the literature includes a large variety of these. A first group of algorithms is made up of those known as "greedy" algorithms, which are based on priority rules or partial enumerative procedures; see Talbot et al. (1986) and more recently Fleszar and Hindi (2003). A second group is composed of enumerative procedures, basically under a *branch and bound* paradigm, Johnson (1988), Hoffmann (1992), Scholl and Klein (1999), Sprencher (2003), currently being the most effective. And a third group composed of applications of diverse metaheuristics (see Scholl and Voss (1996)). Almost all these studies focus on the solution of SALBP-1 o SALBP-2 problems; for this reason, specific procedures must be employed when addressing a problem that includes differences with respect to said models.