Engineering Applications of Artificial Intelligence 49 (2016) 1–9

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

A population-based algorithm for the bi-objective assembly line worker assignment and balancing problem

P. Th. Zacharia, Andreas C. Nearchou*

Department of Business Administration, University of Patras, 265 00 Rio, Greece

ARTICLE INFO

Article history:
Received 11 February 2015
Received in revised form
23 October 2015
Accepted 26 November 2015
Available online 9 December 2015

Keywords:
Assembly line worker assignment and balancing
Genetic algorithms
Multi-objective optimization
Smoothness index
Pareto solutions

ABSTRACT

A multi-objective evolutionary algorithm (MOEA) is presented for the solution of the bi-criteria assembly line worker assignment and balancing problem (ALWABP). This problem consists of determining the best assignment of the assembly tasks to workers as well as the workers to workstations in accordance with some desired objectives. Task times differ depending on worker skills. Two optimization criteria are considered to be minimized, the cycle time and the smoothness index of the workload of the line. The efficiency of the proposed MOEA is evaluated over a set of benchmarks test problems taken from the open literature. A suitable performance analysis is deployed concerning the quality of the Pareto solutions. The results demonstrate a very satisfactory performance in terms of solution quality.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

The assembly line worker assignment and balancing problem (ALWABP) recently introduced by Miralles et al. (2007) is a variant of the simple assembly line balancing problem (ALBP) in which task times are worker dependent. This problem typically occurs in assembly lines with disabled workers in which a worker found efficient to accomplish a particular set of assembly tasks may be inefficient on (or even unable to carry out) another set of assembly tasks. Hence, task times differ depending on worker skills. ALWABP seeks the best assignment of the assembly tasks to workers as well as the workers to workstations in accordance with some desired objectives. Resource allocation involving workers constitutes in general a very complex combinatorial optimization problem. Even without the inclusion of stochasticity the problem is known to be intractable and becomes even harder when skills, shifts and multiple criteria are considered (see the work of De Bruecker et al. (2015) for a recent state-of-the-art survey on this field). Considering ALBPs, Battaïa and Dolgui (2013) discussed and classified multiple variants and extensions of the basic ALBP including ALWABP. Furthermore, recent research (see, for example, the works of Liu et al. (2013); Moreira and Costa (2013); Hemig et al. (2014)) studied interesting real-world ALB models involving workers with multiple skills.

Particularly, Liu et al. (2013) tackled the training and assignment problem of workers in *Seru* production system with the aim of minimizing the total training cost while balancing the total processing times. *Seru* production system is a new type of work-cell based manufacturing environment developed in Japan. Combining the flexibility of job shops with the efficiency of mass production this new production mode showed high success in Japanese electronics manufacturing industry also opening new interesting directions in academic research. Moreira and Costa (2013) examined the problem of balancing assembly lines with heterogeneous workers while considering job rotation schedules. The authors developed a hybrid algorithm which uses heuristics methods together with mixed-integer programs to select the initial solutions and improve them in a post-optimization improvement stage. Hemig et al. (2014) considered an integrated production and staff planning problem occurred in the automotive industry. The authors focused their research on a production environment with heterogeneous parallel assembly lines, and search for a least cost schedule for producing a forecasted demand, taking into account the application of volume flexibility instruments. The problem was modeled by the authors as a non-linear mixed-integer program and solved using dynamic programming.

As in the case of the simple ALBP (see Baybars (1986), Scholl (1999) for a critical survey), four different versions of ALWABP can be defined, termed as: ALWABP-F, ALWABP-1, ALWABP-2, and

* Corresponding author.
E-mail addresses: zacharia@mech.upatras.gr (P.Th. Zacharia), nearchou@upatras.gr (A.C. Nearchou).

http://dx.doi.org/10.1016/j.engappai.2015.11.007
0952-1976/© 2015 Elsevier Ltd. All rights reserved.

Table 1
Summary of previous studies on the ALWABP.

Paper	Problem	Solution method
Miralles et al. (2007)	ALWABP-2	Integer programming model
Miralles et al. (2008)	ALWABP-2	Branch and bound algorithm
Chaves et al. (2007)	ALWABP-2	Clustering search algorithm
Chaves et al. (2009)	ALWABP-2	Clustering search algorithm
Moreira and Costa (2009)	ALWABP-2	Tabu search algorithm
Blum and Miralles (2011)	ALWABP-2	Beam search algorithm
Moreira et al. (2012)	ALWABP-2	Constructive heuristic framework based on priority rules
Moreira and Costa (2013)	ALWABP-2	Hybrid heuristic algorithm and mixed-integer programming
Mutlu et al. (2013)	ALWABP-2	Iterative genetic algorithm
Borba and Ritt (2014)	ALWABP-2	Algorithm 1: randomized beam search Algorithm 2: branch and bound
Vilà and Pereira (2014)	ALWABP-2	Branch and bound algorithm
Moreira et al. (2015)	ALWABP-1	Integer linear model and heuristic approaches

ALWABP-E respectively. The feasibility problem ALWABP-F consists of finding out whether or not a task/worker assignment exist for a given number of m stations, which is feasible for a given cycle time c. That is, ALWABP-F has to decide whether or not a certain combination of values (m, c) is feasible. ALWABP-1 and ALWABP-2 have a dual relationship: the first minimizes m given c; while the second minimizes c given m. As it is obvious, both of these problems require the solution of at least one ALWABP-F instance. ALWABP-E seeks a combination of values (m, c) together with a respective task/worker assignment solution such that the efficiency of the line is maximized.

Within this general background, this paper studies a bi-objective ALWABP aiming to minimize both the cycle time and workload smoothness among the stations. To that purpose, a new heuristic algorithm is presented for the solution of this problem. The new heuristic can be classified as a multi-objective evolutionary algorithm (MOEA). In all our knowledge no previous work in the literature studied ALWABP in the context of multi-objective optimization. This observation stems from a literature review performed with the aim of positioning the developed MOEA within the relevant research field. The outcome of this review is summarized in Table 1 with the papers identified given in chronological order. For each paper, the table shows the specific ALWABP version addressed and the solution method developed. This review allowed us to draw the following general conclusions: (a) All published work concern the solution of the single-objective ALWABP-2 problem. (b) Most of the publications use metaheuristics as a means to determine a near-optimum solution to the problem. Particularly, the following metaheuristic algorithms were developed: clustering search methods, beam search, tabu search, and genetic algorithms. (c) The existing exact solution methods for ALWABP-2 are all based on the branch and bound algorithm. (d) The performance of any new ALWABP solution algorithm is measured over test problems from a set of benchmarks instances developed by Chaves et al. (2007).

The remainder of the paper is organized as follows: Section 2 provides a formal definition of the problem under consideration. Section 3 presents a new population-based heuristic for the solution of the bi-criteria ALWABP. Section 4 describes an example of using the proposed heuristic algorithm for the solution of a particular application problem. Section 5 reports computational results for testing the performance of the algorithm over existing benchmarks test problems taken from the open literature. Finally, conclusions and directions for future work are pointed out and discussed in Section 6.

2. Problem formulation

This section presents the general formulation of the multi-objective ALWABP-2. We make use of the following notation:

n	Number of assembly tasks
m	Number of stations
j	Index for the assembly tasks ($j=1,\ldots,n$)
z	Index for the stations ($z=1,\ldots,m$)
w	Index for the workers ($w=1,\ldots,m$)
t_{jw}	Processing time of task j when executed by worker w
c	Cycle time of the line
SX	Smoothness index
SL_z	Station load of station z; is the set of tasks assigned to station z
ST_z	Station time of station z; is the cumulated task time of the tasks assigned to z
t_{sum}	Total task processing time. $t_{sum}=\sum_{j=1}^{n} t_{jw}$
G	Precedence graph for the assembly tasks
V	The set of vertices in G
E	The set of edges in G

A set of m stations are arranged along an assembly line. Manufacturing a single product on the assembly line requires the partitioning of the total assembly work into a set of n elementary operations called tasks. Each task j ($j=1,\ldots,n$) is performed on exactly one station by a skilled worker w ($w=1,\ldots,m$) and requires a deterministic processing time $t_{jw}>0$ which differs depending on executing operator skills. The tasks are partially ordered by precedence relations defining a precedence graph $G=(V,E)$. G is a directed acyclic graph (DAG) with V being the set of nodes denoting the tasks in G and E the set of edges representing the precedence constraints among the tasks. The assembly line is associated with a cycle time c denoting the maximum (or average) processing time available for each work cycle. Each station z ($z=1,\ldots,m$) can complete its assigned tasks within the specified cycle time c. That is, each station time $ST_z (z=1,\ldots,m)$ should be lower than or equal to c. Under these conditions, ALWABP-2 seeks to determine a *feasible assignment* of the tasks to the stations together with an assignment of the workers to the stations that minimizes c.

To completely define the problem, the following assumptions are made in relation to the line configuration, workforce availability and the tasks operations:

- No buffers are considered in the line.
- A single product is assembled on the line.
- Each worker must be assigned to exactly one station and each station is occupied by only one worker.
- Each task is assigned to only one station.
- No task preemption is allowed.
- Task processing time can be different depending on which worker executes the task.
- A task assignment to the stations is considered to be feasible if: (a) It does not violate the precedence constraints denoted by G; and (b) the resulting station times ST_z do not exceed c.

According to Boysen et al. (2007) classification, ALWABP-2 can be stated as $[pa, link, cum|equip|c]$; where pa denotes that there are processing alternatives in the line (here identical workers

Download English Version:

https://daneshyari.com/en/article/380240

Download Persian Version:

https://daneshyari.com/article/380240

Daneshyari.com