Applied Soft Computing 34 (2015) 655–677

Contents lists available at ScienceDirect

Applied Soft Computing

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

Fuzzy adaptive genetic algorithm for multi-objective assembly line balancing problems

M.H. Alavidoost[a,*], Mosahar Tarimoradi[a], M.H. Fazel Zarandi[a,b]

[a] Computational Intelligent Systems Laboratory, Department of Industrial Engineering, Amirkabir University of Technology, 424 Hafez Avenue, P.O. Box 15875-4413, Tehran, Iran
[b] Knowledge Intelligent Systems Laboratory, University of Toronto, Toronto, Canada

ARTICLE INFO

Article history:
Available online 10 June 2015

Keywords:
Assembly line balancing
Adaptive genetic algorithm
Evolutionary operators
Fuzzy numbers
Fuzzy controller
One-Fifth Success Rule

ABSTRACT

This paper aims at multi-objective straight and U-shaped assembly line balancing problems with the fuzzy task processing times. The problems are referred to herein as f-SALBP and f-SULBP and the objectives that are considered to be satisfied are: (a) minimizing the numbers of stations, (b) maximizing the fuzzy line efficiency, (c) minimizing the fuzzy idleness percentage, and (d) minimizing the fuzzy smoothness index. In fact, the f-SALBP and f-SULBP are SALBP and SULBP generalization in fuzzy circumstance, respectively. Initially, the two problems are formulated and due to the uncertainty, variability and imprecision that often occurred in real-world production systems, the processing time of tasks are supposed as triangular fuzzy numbers. Then, to solve the problem, a hybrid multi-objective genetic algorithm is proposed. A One-Fifth Success Rule (OFSR) is deployed for the selection and mutation operators to improve the genetic algorithm's performance. The results in the genetic algorithm are being controlled in convergence and diversity simultaneously by means of controlling the selective pressure (SP) and mutation rate. Likewise, a fuzzy controller to SP is employed for the OFSR toward a better implementation of the genetic algorithm. In addition, the Taguchi design of experiments is used for parameter control and calibration. Finally, the numerical examples are presented to compare the performance of the proposed method with the existing ones. The results show significantly better performance for the proposed algorithm.

1. Introduction

The competitive market leads producers to promote their manufacturing systems by a more efficient and effective plan in a short period of time. Thus, in the actual design of a manufacturing system, programming an efficient assembly line continuously was an important and controversial issue in the past decades [1]. The manufacturing assembly line was introduced for the first time by *Henry Ford* in the early 1900s [2]. The assembly line balancing problem (ALBP) includes assigning the needed tasks for producing a product as series or batches to a set of stations, so that the objective functions being optimized subject to the limitations [3]. From this point of view, the tasks sequence is the most important issue that should be considered in the assembly line development [4].

There are numerous reviews about ALBP in the literature, and they are generally classified into two main types of Simple ALBP (SALBP) and Generalized ALBP (GALBP). GALBP versions have the extra features such as cost goals, station parallelization, mixed-model production, etc. in comparison with SALBPs [5]. From the goal point of view, SALBP types are divided into SALBP-F, SALBP-1, SALBP-2, and SALBP-E. The SALBP-F is a feasibility problem for a given combination of time cycle and stations number. SALBP-1 and SALBP-2 are the dual of each other, because the SALBP-1 goal is minimizing the station number for a given cycle time, while the SALBP-2 goal is minimizing the cycle time for a given stations number. In the SALBP-E cycle, the time and stations number ought to be minimized simultaneously so that efficiency can be maximized. In addition to the presented classification, the assembly lines can be divided into two categories with respect to their layout, straight assembly lines and U-shaped assembly lines. The straight assembly line is considered as one of the most important traditional mass production sections, and the U-shaped assembly lines are defined to reduce the costs and improve Just-In-Time (JIT) [6]. On the other hand, they can be divided into single models and mixed models with respect to their types of products [3]. In the single model of the assembly line, only one product can be produced in the manufacturing line, and others that can produce more than one product are called mixed model assembly lines. The SALB problem is a single model for straight assembly line balancing and U-shaped layout SALB is called simple U-line balancing (SULB).

* Corresponding author. Tel.: +98 9112116291; fax: +98 1135257759.
E-mail addresses: mosatfa.alavidoost@aut.ac.ir (M.H. Alavidoost), mosahar@aut.ac.ir (M. Tarimoradi), zarandi@aut.ac.ir (M.H.F. Zarandi).

http://dx.doi.org/10.1016/j.asoc.2015.06.001
1568-4946/

The ALBPs were proven to be NP-hard by Gutjahr and Nemhauser [7] and Ajenblit and Wainwright [8]. Therefore, according to the difficulty of such problems, much effort has been made for the development and expansion of heuristic methods such as the Ranked Positional Weighting Technique (RPWT), COMSOAL technique [9], MALB technique [10], MUST technique [11], and LBHA method [12], Critical Path Method (CPM) based approach [13], and also meta-heuristic methods such as genetic algorithm (GA) [8,14], Simulated Annealing (SA) [15], Tabu Search (TS) [16,17], Particle Swarm Optimization (PSO) [18], and Ant Colony Optimization (ACO) [19].

A multi-objective GA for solving the U-shaped assembly line problem was proposed by Hwang et al. [20], and they made a comparison between straight and U-shaped assembly lines. Kim et al. [21] rendered a mathematical model and GA for a two-sided assembly line. In Hwang and Katayama [22] work a multi-decision genetic approach for solving mixed-model U-shaped lines has been proposed which was validated through a case study. A TS algorithm for solving the two-sided assembly line problem was prepared by Özcan and Toklu [23] and the results were benchmarked by the existing approaches. An adaptive GA for solving ALBP was offered by Yu and Yin [24] in which their algorithm efficiency was proven with an example. In another noteworthy work, a hybrid GA was proposed by Akpınar and Mirac Bayhan [25] and deployed for solving the ALB mixed model with parallel station and zoning constraints. Kazemi et al. [26] proposed a two-stage GA for solving mixed-model U-shaped assembly lines. Nearchou [27] used a novel method based on PSO for SALBP and compared it with the existing method. Rabbani et al. [28] proposed a heuristic algorithm based on GA for the mixed-model two-sided assembly line. Chang et al. [29] focused on productivity in the printed circuit board assembly line and rendered a GA with external self-evolving multiple archives in solving this problem. Chutima and Chimklai [30] used a PSO to solve the multi-objective two-sided mixed-model assembly line and showed that if their proposed algorithm was combined with the local search, the quality of its solution set would be better. In another work, Purnomo et al. [31] offered a mathematical model for the two-sided assembly line and solved it with GA and the iterative first-fit rule method, and finally compared the results of these methods. Manavizadeh et al. [32] proposed an SA for a mixed model assembly U-line balancing type-I problem and compared the algorithm results with the exact method. Hamzadayi and Yildiz [33] used an SA algorithm for the line balancing problems and modeled sequencing in U-shaped assembly lines. Dou et al. [34] proposed a discrete PSO for solving SALBP-1 and compared their results with GA. Kalayci and Gupta [35] used a PSO with a neighborhood-based mutation operator for solving the sequence-dependent disassembly line balancing. Zha and Yu [36] proposed a hybrid ant colony algorithm for solving the U-line balancing and rebalancing the problem and compared their algorithm results with the existing methods. Among these meta-heuristic methods, most of the studies were devoted to GA and these previous research studies have indicated that there must be sufficient motivation to use this popular algorithm for solving the emerging problem. To perform a controlled random search for identifying the optimal solution, an alternative traditional optimal technique was provided in the complex circumstances [37]. The focus of many researchers on GA and its popularity was the authors' motivation to improve the performance of this meta-heuristic through a modification as a part of the contribution of this paper and put it into practice to solve the mentioned controversial problem.

Numerous works have been reviewed that have solved ALBPs in crisp circumstances whilst actual world problems usually deal with uncertainty and vagueness. To represent uncertainty, fuzzy numbers can reflect the ambiguity of real data well. Considerable attention has been given to ALBPs in the literature, only some of which have managed to solve such problems in the fuzzy environment. In other words, in comparison with crisp ALBPs, few researchers have focused on fuzzy ALBP so far [37,38]. Among the articles focusing on solving the fuzzy ALBP by precise methods, few researches [39–42] are noticeable.

Studies in this area reveals those which have used heuristic and meta-heuristic methods for solving the ALBP in a fuzzy environment are rare. In the 90s Tsujimura et al. [43] and Gen et al. [44] initialized using fuzzy GA for this problem. With a typical GA provided that the tasks processing time was presented in fuzzy numbers, they solved SALBP-1. While Brudaru and Valmar [45] proposed a combined GA with a Branch and Bound method to solve SALBP-1. Fonseca et al. [2] presented and modified the Ranked Positional Weighting Technique and COMSOAL method with fuzzy numbers, and applied it to solve these sort of problems. Hop [46] proposed a heuristic method to solve a fuzzy mixed-model ALBP. Zhang et al. [47] prepared a heuristic method to solve SULBP with fuzzy numbers. Özbakır and Tapkan [48] presented a model for two-sided ALBP and solved this problem by Bees Algorithm. Zacharia and Nearchou [49] also introduced a multi-objective GA to solve SALBP-2 with fuzzy numbers, in which they applied the weighted sum of objectives. Zacharia and Nearchou [50] presented a meta-heuristic algorithm based on the genetic algorithm for solving SALBP-E.

As mentioned, since numerous researchers used GA and its popularity, this paper tends to improve the performance of this algorithm through a modification. Likewise, it is noteworthy that no research has considered and solved SULB-1 using meta-heuristic methods in fuzzy circumstances. So this paper has considered the SALB-1 and SULB-1 in which a modified GA is presented with the OFSR that results in enhancing the performance. A fuzzy controller for better adaptation between GA and the OFSR has been rendered and also the parameters of the proposed algorithm have been calibrated by the Taguchi design of experiments. Due to the uncertainty in the real world, fuzzy numbers have been used to represent the assembly line cycle and processing time.

The rest of the paper is organized as follows: in Section 2, the main characteristics of SALBP and SULBP are represented. In Section 3, the fuzzy arithmetic is provided as well as a number of criteria to sort the fuzzy numbers. To present the contribution of the genetic algorithm, OFSR and also the procedure of genetic algorithm modification with OFSR are presented in Section 4. In Section 5, the parameters of the proposed algorithm will initially be calibrated using the Taguchi method, and then the proposed algorithm will be examined by benchmarks and its results will be compared with the existing methods. Finally, conclusions and some guidelines for future studies are provided in Section 6.

2. Problem formulation

This section represents the main characteristics of SALBP-1 and SULBP-1. As mentioned before, the assembly line is a series of locations which are called stations, and a subset of tasks that are performed and need to be processed for the production of a unit in these locations [44]. For these problems, the available information is as follows [51]:

- A given set of tasks $J = \{i | i = 1, 2, \ldots n\}$.
- The set of tasks' needed time which is shown as $T = \{\tilde{t}_i | i = 1, 2, \ldots n\}$.
- Each task's allocated time that will be presented as triangular fuzzy number (TFN).
- The set of precedence relations $P = \{(a, b | \text{task a must be completed before task} b)\}$.
- Maximum allowed fuzzy cycle time ($\tilde{C}_{max}$).