Computers & Operations Research 40 (2013) 2556–2563

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

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

Two approaches for solving the buffer allocation problem in unreliable production lines

Leyla Demir [a,*], Semra Tunalı [b], Deniz Türsel Eliiyi [c], Arne Løkketangen [d]

[a] Pamukkale University, Department of Industrial Engineering, 20070 Denizli, Turkey
[b] Izmir University of Economics, Department of Business Administration, Sakarya Cad. No: 156, Balcova-Izmir, Turkey
[c] Izmir University of Economics, Department of Industrial Systems Engineering, Sakarya Cad. No: 156, Balcova-Izmir, Turkey
[d] Department of Informatics, Molde College, Molde, Norway

ARTICLE INFO

Available online 17 February 2013

Keywords:
Buffer allocation problem
Production lines
Tabu search
Combinatorial optimization

ABSTRACT

This paper presents an integrated approach to solve the buffer allocation problem in unreliable production lines so as to maximize the throughput rate of the line with minimum total buffer size. The proposed integrated approach has two control loops; the inner loop and the outer loop. While the inner loop control includes an adaptive tabu search algorithm proposed by Demir et al. [8], binary search and tabu search are proposed for the outer loop. These nested loops aim at minimizing the total buffer size to achieve the desired throughput level. To improve the efficiency of the proposed tabu search, alternative neighborhood generation mechanisms are developed. The performances of the proposed algorithms are evaluated by extensive computational experimentation, and the results are reported.

1. Introduction

A production system that is composed of serially connected machines with buffers in between is called a production line, where each part goes through all machines in exactly the same order and waits in buffer areas for the next operation. The flow of parts may be disrupted by machine failures or variable processing times. The efficiency of such a production line can be improved by distributing buffers between machines so as to minimize the disruptions.

The buffer allocation problem is an NP-hard combinatorial optimization problem [23,27], which deals with finding optimal buffer sizes to be allocated into buffer areas in a production line. In general, the buffer allocation problem is classified into three categories according to its objective function. While the first problem attempts to maximize the throughput rate of the line with a fixed amount of buffer size, the second problem aims to minimize the total buffer size to achieve a desired throughput rate, and the last problem aims to minimize the average work-in-process in the line.

Due to its complexity and importance, the buffer allocation problem has been studied widely and numerous publications are available in the literature. A comprehensive literature survey on buffer allocation problem can be found in Demir et al. [9]. Here, we summarize some recent studies. Most of the literature on buffer allocation focuses on the first problem while few studies address the other two problems. Gershwin and Schor [14] name the first problem as the dual problem and the second as the primal problem. The authors propose gradient-based search algorithms for solving both problems in unreliable lines while using the solution of the dual problem to solve the primal problem. Yamashita and Altiok [45] propose a dynamic programming algorithm for minimizing the total buffer size in reliable lines with phase-type processing times. Lutz et al. [26] use tabu search for maximizing the throughput rate of a reliable line while minimizing the total buffer size in the system. Gurkan [17] employs sample path optimization and stochastic approximation for minimizing the total buffer size in unreliable continuous production lines where the product type is fluid. Han and Park [18] present an approximation method to find the optimal buffer sizes in unreliable lines with quality inspection machines so as to achieve the desired throughput rate. Tempelmeier (2003) [44] optimizes the buffer and workload allocation simultaneously by using a multi-objective approach in unreliable lines. The Powell's algorithm is employed to solve the buffer allocation problem in general configured networks for buffer size minimization by MacGregor Smith and Cruz [27]. Later, Cruz et al. [6] propose a solution approach based on Lagrangian relaxation for the same problem. In a more recent paper, Battini et al. [4] propose a new paradigm – the buffer design for availability – to solve the buffer allocation problem. For this purpose, simulation is used to describe the effects of workstation reliability parameters on buffer capacities while minimizing the total buffer size. Demir et al. [7] propose a tabu search algorithm to minimize the total

* Corresponding author. Tel.: +90 258 296 3095.
E-mail address: ldemir@pau.edu.tr (L. Demir).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.02.004

buffer size. The authors show the efficiency of their proposed algorithm on previously published benchmark problems.

In this study, an integrated approach involving two control loops is proposed for solving the buffer allocation problem. While the inner loop includes an adaptive tabu search (ATS) algorithm by Demir et al. [8] to obtain the maximum throughput rate of the line for a given total buffer size, two different algorithms, i.e., binary search and tabu search are proposed in the outer loop to minimize the total buffer size in the system so that the desired throughput rate can be achieved. To the best of our knowledge this is the first study to integrate two different algorithms for minimizing total buffer size. Previous studies hybridizing two optimization methods consider the problem only from a throughput maximization perspective as in the studies of [40] and [12].

The rest of this paper is organized as follows. The next section explains problem specifications. The proposed integrated solution approach is described in Section 3. The computational results on the performance of the proposed algorithms are presented in Section 4. Finally in Section 5, concluding remarks and some future research directions are given.

2. Problem specifications

We consider the buffer allocation problem in an unreliable serial production line. The characteristics of the line can be listed as follows:

- Each part goes through all machines in exactly the same order.
- There is an intermediate location for storage (buffer) between each pair of machines.
- Machines in the line have unique deterministic processing times.
- Machines are subject to breakdown, and the repair and failure rates are geometrically distributed.
- The first machine is never starved, i.e., input is always available, and the last machine is never blocked, i.e., there is always space to dispose of the output.

Assuming there are K machines and $K-1$ buffers in a production line, our objective is to minimize the total buffer size so as to achieve the desired throughput rate. The problem can be formulated as follows:

Find $B=(B_1,B_2,\ldots,B_{K-1})$ so as to

$$\min N=\sum_{i=1}^{K-1} B_i \quad (1)$$

subject to

$$f(B^N)\geq f^* \quad (2)$$

$$0\leq B_i\leq u_i \quad (3)$$

$$B_i \text{ nonnegative integers } (i=1,2,\ldots,K-1) \quad (4)$$

where K is the number of machines in the line, B is the buffer vector, N is the total buffer size, u_i is the upper bound for each location, $f(B^N)$ is the throughput rate of the production line obtained when total buffer size is N, and f^* is the desired throughput rate. It should be noted that upper bounds (u_i) for each buffer location are chosen such that their summation is larger than the total buffer size (N) in the system. This problem is hard because constraint 2 cannot be expressed in closed form. Consequently, even if we know one buffer vector B satisfying $f(B^N)=f^*$, it is hard to construct another.

In general, solution approaches for buffer allocation problem involve a procedure where generative methods and evaluative methods are combined in a closed loop configuration. The evaluative method is used to obtain the performance measures of the production line. These performance measures are then communicated to the generative method for optimizing the buffer configuration. Traditional Markov state models [20,21,33,34], aggregation method [10–12,35], generalized expansion method [1,6,27], decomposition method [7,8,14,19,28,30,39,40,44], and simulation ([2,4,17,36]) are examples of evaluative methods. Due to its ability to obtain the throughput of a production line quite accurately and quickly, the decomposition method proposed by Gershwin [13] is employed as an evaluative method in this study. The interested reader is referred to Demir et al. [7,8] for details of the decomposition method and its application on the buffer allocation problem.

Various generative methods are used for solving the buffer allocation problem including dynamic programming [5,10,24,45]; Powell's method ([27,46]), Lagrange-based methods [6], nonlinear programming [39], gradient-based search methods [14,19,22, 37,38], Hooke and Jeeves method [3], and the degraded ceiling method [30]. Recently, some meta-heuristics are also employed effectively for solving the buffer allocation problem. Simulated annealing [41–43], tabu search [7,8,26,40], genetic algorithm [11,12,42], and ant colony optimization [31,32] are examples in this area.

Unlike the above studies that address the problem from a throughput maximization perspective our study integrates different algorithms in a nested loop, and approaches the problem from a total buffer size minimization perspective while maximizing the throughput rate for each level of the total buffer size. The next section explains the proposed approach for this integrated problem in detail.

3. Solution approach

Fig. 1 summarizes the execution mechanism of the proposed approach. The outer loop algorithm is started with a pre-specified N value for total buffer size, and the maximum throughput rate that can be obtained with this N value is computed via ATS [8]. This throughput rate is then compared to the desired f^*, and new N values are suggested by the outer loop algorithm in an iterative manner. The procedure continues until the termination criterion of the outer loop is satisfied, i.e., until the desired throughput

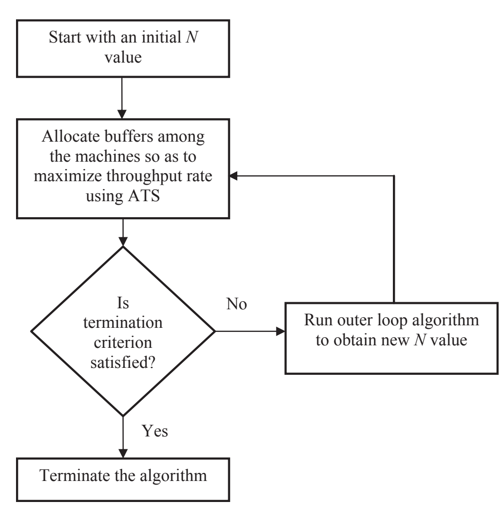

Fig. 1. The framework of the integrated approach for total buffer size minimization.