Swarm and Evolutionary Computation 28 (2016) 131–143

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

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

Regular Paper

A new cuckoo search algorithm for 2-machine robotic cell scheduling problem with sequence-dependent setup times

Arindam Majumder, Dipak Laha *

Department of Mechanical Engineering, Jadavpur University, Kolkata, India

ARTICLE INFO

Article history:
Received 8 July 2015
Received in revised form
28 October 2015
Accepted 8 February 2016
Available online 11 March 2016

Keywords:
Scheduling with sequence-dependent setup times
2-Machine centered robot cell
One-unit cycles
Total cycle time
Cuckoo search algorithm
Minimal part set sequence
Sequence of robot moves

ABSTRACT

The paper addresses the problem of 2-machine robotic cell scheduling of one-unit cycle with sequence-dependent setup times and different loading/unloading times of the parts. As an alternative metaheuristic algorithm, the cuckoo search algorithm has recently attracted growing interests of researchers. It has the capability to search globally as well as locally to converge to the global optimality by exploring the search space more efficiently due to its global random walk governed by Levy flights, rather than standard isotropic random walk. In this study, a discrete cuckoo search algorithm is proposed to determine the sequence of robot moves along with the sequence of parts so that the cycle time is minimized. In the proposed algorithm, the fractional scaling factor based procedure is presented to determine the step length of Levy flights distribution in discrete from and then, using this step length, two neighborhood search techniques, interchange and cyclical shift methods are applied to the current solution to obtain improved solution. A response surface methodology based on desirability function is used to enhance the convergence speed of the proposed algorithm. Also, a design of experiment is employed to tune the operating parameters of the algorithm. Finally, empirical results with a large number of randomly generated problem instances involving large part sizes varying from 200 to 500 under different operating conditions are compared with two well-known algorithms in the literature and demonstrate the effectiveness of the proposed algorithm.

1. Introduction

The problem of cyclic robotic cell scheduling has drawn a lot of attention among the researchers [10,15,31,34,38,59,6,60,46] due to its complexity as well as its immense importance in modern manufacturing industries. Typical applications of robots that are frequently encountered in advanced manufacturing environment include semiconductor manufacturing, automobiles, engine block manufacturing, and textile mills industries [39]. A robotic cell consists of an input station, a number of machines arranged in series, an output station and one or more robots for handling the parts between the stations and machines [19]. The cyclic robotic cell scheduling problem deals with the scheduling of a minimum set of parts to be processed on different machines and this schedule of parts is repeated to meet the total demand of the parts. Since the cyclic robotic cells scheduling or cyclic hoist scheduling problems are usually studied based on deterministic and static production conditions in the literature [30–32,37,40], a deterministic and static two-machine robotic cell scheduling problem of one-unit cycle with sequence-dependent setup times is considered in this study. The objective of the problem is to determine the optimal sequence of robot moves along with the sequence of parts in order to minimize the total cycle time. Each of the repeated sequence of parts is called a cycle and the total time taken by a robot for completing one cycle is known as the total cycle time.

Since the robotic cell scheduling problem belongs to NP-hard class in the strong sense [59], exhaustive enumeration of all possible solutions is computationally expensive. Therefore, different optimization techniques such as exact optimization, heuristics, and metaheuristics have been proposed to solve these problems. There have been a number of studies based on the exact optimization methods like mixed integer programming [23,36,41,44,13,53] and branch and bound method [11,14,35,54] to demonstrate the superior performance of these techniques, but these methods are suitable only for small-sized and medium-sized problems due to their exponential time requirements. Apart from the exact optimization methods, some researchers [33,57,45] proposed heuristic algorithms for the cyclic robotic flowshop scheduling problems. However, these algorithms provide good near-optimal solutions in some special cases for large-sized problems. On the other hand, metaheuristic algorithms are robust and adaptive search optimization methods and can be

* Corresponding author.
E-mail addresses: arindam2012@gmail.com (A. Majumder), dipaklaha_jume@yahoo.com (D. Laha).

http://dx.doi.org/10.1016/j.swevo.2016.02.001
2210-6502/

used to generate optimal or near-optimal solutions for large-sized scheduling problems in reasonable computational times.

There have been growing number of studies on different types of robotic cell scheduling problems in the literature. Sethi et al. [49] studied a robotic cell scheduling problem to determine sequencing parts and robot move in order to maximize the long-run average production rate of each part type to be processed in the cell. Later on, a considerable number of studies have focused on modeling and optimization of different robotic cell scheduling [1,12,21,26, 3,5,7]. Recently, the robotic cell scheduling problems have been solved using metaheuristic algorithms like particle swarm optimization [28,29], genetic algorithm [17,2], and simulated annealing [24].

Although there have been several studies on the complexities, various exact optimization techniques such as branch and bound method, dynamic programming, mixed integer programming, and few polynomial algorithms in some special cases, modeling of the robotic cell scheduling problem with sequence-dependent setup times using metaheuristic algorithms except simulated annealing [59] was not reported in the literature. There are noteworthy metaheuristic algorithms, namely, genetic algorithms, simulated annealing, differential algorithm, and ant and bee algorithms. Recently, cuckoo search algorithm has emerged a promising metaheuristic algorithm and have been utilized as successful optimization method in various optimization problems. The cuckoo search algorithm has recently attracted growing interests of researchers due to its capability to search globally as well as locally to converge to the global optimality by exploring the search space more efficiently for its global random walk governed by Levy flights, rather than standard isotropic random walk. Since Levy flights have infinite mean and variance, this algorithm is more efficient than algorithms by standard Gaussian process. Recent studies have shown that this algorithm is potentially more efficient than many existing metaheuristic algorithms such as particle swarm optimization algorithm, artificial bee colony algorithm, and differential evolution [56]. This algorithm has been successfully applied in different engineering and management problems such as wireless sensor networking [20], scheduling [18,27,42,9], machining operations [58], traveling salesman [43], reliability optimization [51], structural optimization problems [22,25], and video target tracking [52].

Therefore, the research is motivated to substantiate the suitability of potential applications of cuckoo search algorithm using the problem specific information and to investigate the comparative performance of this algorithm with some best-known metaheuristics, especially for solving large sized robotic cell scheduling problems. Inspired by this motivation, this paper presents a discrete cuckoo search algorithm to solve this robotic cell scheduling problems. In the proposed algorithm, a methodology is applied to determine the step length of the Levy distribution and then, using this step length, two neighborhood search techniques, interchange and cyclical shift methods are used to the current solution to obtain improved solution. The exhaustive computation experimentation of the proposed algorithm with respect to lower bounds considering a set of large-sized problems under different operating conditions is carried out.

The remaining of the paper is organized as follows: Section 2 describes the problem definition and formulation. Section 3 presents the proposed discrete cuckoo search algorithm. The detailed tuning of the process parameters used in the algorithm are presented in Section 4. Section 5 discusses the computational results of the proposed algorithm and its comparison with some noteworthy existing algorithms. Finally, in Section 6, the conclusions are drawn.

2. Problem definition and the objective function

Consider the problem of 2-machine robotic cell scheduling of one-unit cycle with sequence-dependent setup times as well as with different loading / unloading time for each part. Consider a centered robot cell with two machines, single-gripper robot, one input (I) and one output drive (O) as shown in Fig. 1. Let k represent different part types, $N=d_1+d_2+\ldots+d_k$ total demand of parts, $n=n_1+n_2+\ldots+n_k$ total number of parts in the minimal part set (MPS), where, d_k and n_k are the demand of parts of type k and minimum ratio of parts of type k in one MPS. Therefore, in order to meet the total demand of N parts, MPS is repeatedly manufactured r times (here, r short cycles), where, r is the greatest common divisor of $d_1, d_2, \ldots, d_k$ demands and $N=rn$. The problem is to determine the sequence of robot moves and the MPS sequence of n parts of k types for the 2-machine robotic cell scheduling problem so that total cycle time is minimized. Here, MPS sequence refers to a solution which is one sequence among the total number of feasible sequences, i.e., $n!/\prod_{s=1}^{k}(n_s!)$ Over MPS. It is assumed that the loading time of each part is dependent upon the processing of the immediate preceding part in the sequence of parts and there is no buffer storage between the machines.

For the 2-machine robotic cell scheduling problem, the robot can move into two possible sequences, namely, S_1 and S_2. If the initial state starts from the loading of i-th part on m_2, then the sequences (S_1 and S_2) can be described as follows:

- S_1: The robot waits at m_2 until i-th part has been processed, unloads the i-th part from m_2, moves to O, drops the part at O, moves freely to I, picks up $(i+1)$-th part, moves to m_1, unloads the $(i+1)$-th part on m_1, waits at m_1 until the $(i+1)$-th part has been processed at m_1, unloads the $(i+1)$-th part from m_1, moves to m_2 and loads the $(i+1)$-th part on m_2.
- S_2: The robot moves freely to I after loading i-th part on m_2, picks up $(i+1)$-th part from I, moves to m_1, loads $(i+1)$-th part on m_1, moves freely to m_2, waits at m_2 (if necessary) until i-th part is being processed at m_2, unloads the i-th part from m_2, moves to O, drops the i-th part at O, moves freely to m_1, waits at m_1 (if necessary) until the $(i+1)$-th part is being processed at m_1, unloads the $(i+1)$-th part from m1, moves to m_2 and loads $(i+1)$-th part on m_2.

Let the time interval between loading the part in the i-th position of the MPSS on m_2 machine and loading the part in (i+1)-th position of the MPSS on m_2 machine for the robot move sequence S_1 and S_2 be $T_1(\sigma_i, \sigma_{i+1})$ and $T_2(\sigma_i, \sigma_{i+1})$. These partial cycle times as given by Zarandi et al. [59] are obtained from the

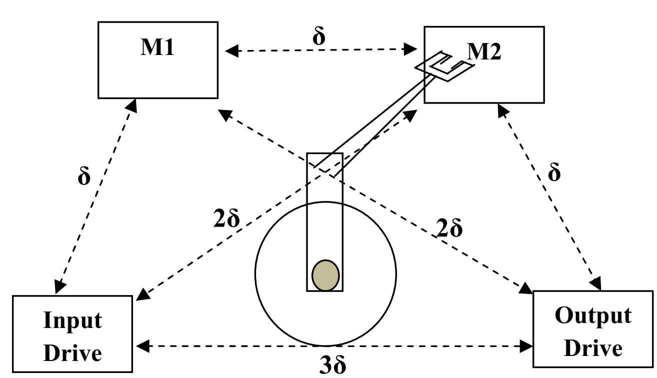

Fig. 1. A centered robot cell with two machines, one input and one output drive.