Applied Mathematics and Computation 248 (2014) 143–159

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

An effective differential evolution algorithm for permutation flow shop scheduling problem

Ying Liu [a], Minghao Yin [c,*], Wenxiang Gu [b,c]

[a] College of Humanities & Science, Northeast Normal University, Changchun 130117, China
[b] Department of Basic Subjects Teaching, ChangChun Architecture & Civil Engineering College, ChangChun 130607, China
[c] School of Computer Science and Information Technology, Northeast Normal University, Changchun 130117, China

ARTICLE INFO

Keywords:
Permutation flow shop problem
Differential evolution
Individual improving scheme
Greedy-based local search
Scheduling
Swarm intelligence

ABSTRACT

The permutation flow shop problem (PFSSP) is an NP-hard problem of wide engineering and theoretical background. In this paper, a hybrid differential evolution (DE) called L-HDE is proposed to solve the flow shop scheduling problem which combines the DE with the individual improving scheme (IIS) and Greedy-based local search. First, to make DE suitable for PFSSP, a new Largest-Rank-rule (LRV) based on a random key is introduced to convert the continuous position in DE to the discrete job permutation so that the DE can be used for solving PFSSP. Second, the NEH heuristic was combined with the random initialization to initialize the population with certain quality and diversity. Third, the IIS-based local search is used for improving the diversity of population and enhancing the quality of the solution with a certain probability. Fourth, the Greedy-based local search is designed to help the algorithm to escape from local minimum. Additionally, simulations and comparisons based on PFSSP benchmarks are carried out, which show that our algorithm is both effective and efficient.

1. Introduction

Scheduling problems play an important role in both manufacturing systems and industrial processes for improving the utilization of resources, and therefore it is important to develop effective and efficient scheduling approaches to find the optimal solutions [1]. As a simplified version of the flow shop problem, The permutation flow shop problem (PFSSP), one of the best known problems in the area of scheduling, has been proved to be non-deterministic-polynomial-time (NP)-hard [2,3]. Due to its significance in both academic and engineering applications, it has gained wide research work.

Since the pioneering work of Johnson [4], many methods have been introduced for solving PFSSP with the objective of minimizing the makespan. However, due to unacceptable computation time, exact algorithms such as the branch and bound method [5–8] and the mixed integer linear programming method [9] can only solve problems with relatively small-sized problems. Heuristic algorithms were then proposed to solve the large-sized scheduling problems. The heuristic algorithms seem to have better performance than other algorithms. The heuristic algorithms can be broadly classified into three categories: constructive heuristic algorithms, improvement heuristic algorithms and hybrid heuristic algorithms [10]. The constructive heuristics are mainly simple heuristics, as is seen in Palmer [11], Gupta [12], Ho and Chang [13], Campbell [14],

* Corresponding author.
E-mail address: ymh@nenu.edu.cn (M. Yin).

http://dx.doi.org/10.1016/j.amc.2014.09.010
0096-3003/

Rajendran [15], Nawaz et al. [16], Taillard [17], and Framinna et al. [18]. Constructive heuristics usually can obtain a nearly optimal solution in a reasonable computational time, although some of them are time-consuming.

The improvement heuristics are mainly meta-heuristics which start from previous generated solutions and subsequently approach the optimal solution by improving the solutions with domain dependent knowledge. The meta-heuristics mainly include simulated annealing algorithm (SA) [19,20], genetic algorithm (GA) [21,22], Artificial Immune system Algorithm (AIS) [23], particle swarm optimization algorithm (PSO) [24,44,45], ant colony algorithm (ACO) [25,26], tabu search algorithm [27–30], iterated local search algorithm (ILS) [31], and estimation of distribution algorithm (EDA) [32].

Rather recently, it has become evident that the concentration on a sole meta-heuristic has some limitations. Researchers have found that a skilled combination of two or more meta-heuristic techniques, called hybrid heuristics, may improve the performance especially when dealing with real-world and large scale problems. A lot of hybrid heuristic based algorithms have been investigated in the past few years. In [33], a hybrid SA algorithm was introduced combining the operators of GA with local searches. In [34], the genetic algorithm is integrated into a novel local search scheme resulting in two hybrid algorithms: the insertion search and the insertion search with cut-and-Repair (ISCR). In [35], two effective heuristics are used during the local search to improve all generated chromosomes in every generation. In [36], Wang Ling used the well-known Nawaz–Enscore–Ham (NEH) combined with GA to generate the initial population, and applied multi-crossover operators to enhance the exploring potential. Murata proposed a hybrid genetic algorithm with local search in [37]. In [24], Tasgetiren et al. applied the PSO algorithm to solve PFSSP by using a small position value rule, and the proposed algorithm, as called PSOVNS, was combined with the variable neighborhood-based local search algorithm. Liu, in [10], proposed an efficient particle swarm optimization based mimetic algorithm (MA) for PFSSP to minimize the maximum completion time. In this algorithm, the PSO-based searching operators and some special local search operators are used to balance the exploration and exploitation abilities. In [26], an ant-colony based algorithm was proposed to solve PFSSP, combining the fast local search to enhance the solutions. In [38], a hybrid algorithm that combined Framinan–Leisten (FL) heuristic with iterated local search algorithm is proposed. Bassem and Eddaly applied the EDA algorithm to solve PFSSP by using a small position value rule, and the new algorithm, as called EDAVNS, was combined with the variable neighborhood search algorithm as an improvement procedure after creating a new offspring [32].

Among the existing meta-heuristic algorithms, a new evolution technique, differential evolution, is a population-based heuristic evolutionary algorithm that is simple to be implemented and has little or no parameters to be tuned. One of the remarkable advantages of DE is that this algorithm can use mutation, cross, and selection operators to increase the population diversity. Up to now, most published works on DE mainly focused on solving the complex continuous optimization problem. Within our knowledge, only few of researchers have used the DE algorithm to solve PFSSP. In 2004, Tasgetiren proposed a DE based algorithm to deal with PFSSP by using the SPV rule [39]. In [40], a DE based algorithm was proposed for solving PFSSP when all jobs are available for processing and the objective is to minimize the makespan. In [41], in order to apply the DE algorithm to solve PFSSP, a largest-order-value rule was introduced and a small but efficient local search was designed to enhance the quality of the solution. However, this field of study is still in its early days, and a large number of future researches are necessary in order to develop DE based algorithms for solving PFSSP other than only for those areas the inventors originally focused on.

In this paper, we propose a new hybrid DE (L-HDE) algorithm combining DE with some local search mechanisms as well as an individual improvement scheme (IIS) for solving PFSSP. The crucial idea behind L-HDE can be summarized as follows. Firstly, to make DE suitable to solve PFSSP, a new LRV rule is proposed based on a random key. This rule can help to convert the continuous encoding of DE to a job permutation. Secondly, The NEH heuristic was combined with the random initialization to initialize the population with certain quality and diversity. Thirdly, the individual improvement scheme (IIS) is proposed as a component of DE, for the sake of improving DE's local search performance. Fourth, Greedy-based local search is designed and incorporated as a local search scheme into the searching process to enrich the searching behaviors and to avoid premature convergence. Experimental results and comparisons have shown that our algorithm is effective and efficient.

The rest of this paper is organized as follows. In Sections 2 and 3, we will introduce PFSSP and DE respectively. In Section 4, the L-HDE algorithm is proposed after presenting solution representations, DE-based search, IIS-based local search, and Greedy-based local search. The experimental results of the L-HDE and comparisons to other previous algorithms are shown in Section 5. In the last section we conclude this paper and point out some future work.

2. Permutation flow shop scheduling problem

The permutation flow shop scheduling problem (PFSSP) in the paper consists of a set of jobs on a set of machines with the objective of minimizing the makespan. In permutation flow shop problem, n jobs $N = J_1, J_2, \ldots, J_n$ are to be processed on a series of m machines $M = M_1, M_2, \ldots, M_n$ sequentially. Each job consists of a set of operations $J_i = \{O_{i1}, \ldots, O_{jm}\}$. The processing time of job J_i on machines M_i is denoted by $P_{i,j}$ ($i = 1, \ldots, m, j = 1, \ldots, n$). Each job can be processed on only one machine at a time and each machine can process only one job at a time. Moreover, the operation cannot be preemptable. The sequence in which the jobs to be processed are identical for each machine. The objective of the scheduling is to find the minimum makespan.