Applied Mathematics and Computation 243 (2014) 637–643

Contents lists available at ScienceDirect

Applied Mathematics and Computation

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

Security based bi-objective flow shop scheduling model and its hybrid genetic algorithm

Ren Qing-dao-er-ji [a], Yuping Wang [b,*]

[a] School of Information Engineering, Inner Mongolia University of Technology, Hohhot 010051, China
[b] School of Computer Science and Technology, Xidian University, Xi'an 710071, China

ARTICLE INFO

Keywords:
Multi-objective flow shop scheduling
Genetic algorithm
Crossover operator
Mutation operator
Local search

ABSTRACT

In this paper, we considered the flow shop scheduling problem with respect to the both objectives of the makespan and the mean continuous running time, proposed a security based bi-objective flow shop scheduling model. To solve the proposed model more effectively, we presented a hybrid genetic algorithm (HGA), which used some tailor made genetic operators and a local search operator in order to improve the local search ability of GA. The proposed algorithm is tested with some well-known problems in literature. The computational results demonstrated the effectiveness of the proposed algorithm.

1. Introduction

The flow shop scheduling problem (FSSP) has an extensive background in manufacturing systems and has attracted many researchers' attention since it was proposed by Johnson [1]. Many researches for single objective FSSPs result in a schedule to minimize the makespan. The conventional approaches to solve single-objective FSSP can be mainly divided into two categories, namely, exact and approximation methods. Exact methods [2,3], such as linear programming, branch and bound and Lagrangian relaxation, have been successfully used to solve small-sized FSSPs. However, they are still incapable of solving large-scale real-world FSSPs. For the large-scale problems, approximation methods are superior to the exact methods. These kinds of methods mainly include genetic algorithms (GA), simulated annealing (SA), ant colony optimization (ACO), particle swarm optimization (PSO) and tabu search (TS).

However, purely single objective FSSPs cannot totally reflect the needs of the real-world practical applications. Multi-objective FSSPs (MOFSSP) should be the trend in the future for the real-life scheduling problem. Therefore, few researchers considered the FSSP as a multi-objective problem and solved. Gupta et al. [4] considered the two-machine FSSP where it was desired to find a minimum total flow time schedule subject to the condition that the makespan of the schedule was minimum. T'kindt et al. [5] considered the 2-machine FSSP with the objective of minimizing both the total completion time and the makespan criteria. Ravindran et al. [6] developed some multi-criterion approaches to FSSPs by considering makespan time and total flow time. Eren et al. [7] considered a bi-criteria scheduling problem with sequence-dependent setup times on a single machine. The objective function of the problem was minimization of the weighted sum of total completion time and total tardiness. Pasupathy et al. [8] proposed a multi-objective genetic algorithm named Pareto GA with an archive of non-dominated solutions subjected to a local search. Lemesre et al. [9] proposed a parallel exact method to solve bi-objective permutation FSSP. Li and Wang [10] solved MOFSSP using a hybrid quantum-inspired GA. Chang et al. [11] presented a

* Corresponding author.
E-mail addresses: renqingln@sina.com (R. Qing-dao-er-ji), ywang@xidian.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.amc.2014.06.019
0096-3003/

mining gene structures on subpopulation genetic algorithm which was combined with mining gene structure approach and subpopulation genetic algorithm. Tseng et al. [12] considered an n-job, m-machine lot-streaming problem in a FSSP with equal-size sub lots where the objective was to minimize the total weighted earliness and tardiness. Naderi et al. [13] applied a novel simulated annealing to hybrid FSSP to minimize both total completion time and total tardiness. Dugardin et al. [14] focused on the multi-objective resolution of a reentrant hybrid FSSP. Karimi et al. [15] presented a multi-phase approach to tackle hybrid flexible FSSP considering the minimization of makespan and total weighted tardiness simultaneously. Josef Geiger et al. [16] described the proposition and application of a local search meta heuristic for multi-objective optimization problems. Chiang et al. [17] considered the makespan and the total flow time as objectives, and proposed a memetic algorithm to solve the FSSP. Dubois-Lacoste et al. [18] presented a new algorithm for five bi-objective permutation FSSP that arise from the pair wise combinations of the objectives: makespan, the sum of the completion times of the jobs, and the weighted and non-weighted total tardiness of all jobs. Cho et al. [19] dealt with a scheduling problem for reentrant hybrid FSSP with serial stages where each stage consists of identical parallel machines.

From the aforementioned literatures, we can know that the makespan, tardiness time, lateness, earliness and flow time are often used as the optimization criteria in MOFSSP. All of these objectives are constructed based on the completion situation of the jobs and are without taken the situation of the machines into account. In this paper, we considered both the safely usage of the machines and the completion situation of the jobs, constructed a security based MOFSSP model to minimize the makespan and the mean continuous running time. The detailed contributions of this paper are as follows:

(1) We constructed a MOFSSP model to minimize the makespan and the mean continuous running time.
(2) We used tailor made crossover operator and mutation operator to improve the ability of GA.
(3) A local search operator was designed so as to improve the local search ability of GA.
(4) Based on these genetic operators, we proposed a hybrid genetic algorithm (HGA). Finally, the efficiency of the proposed algorithm was verified by computer simulations on some typical scheduling problems.

The remainder of the paper is organized as follows. We first discuss about a multi-objective problem in Section 2. Then, we describe the FSSP and its mathematical model in Section 3. In Section 4, a hybrid genetic algorithm to the MOFSSP is presented. Section 5 presents the experimental results. The conclusions are made in Section 6.

2. Multi-objective optimization and pareto optimality

We can describe a multi-objective optimization problem with k objectives as follows [20]:

$$\text{Minimize } y = f(X) = [f_1(X), f_2(X), \ldots, f_k(X)]. \tag{1}$$

$$\text{Subject to } g_i(X) \geqslant 0 \quad i = 1, 2, \ldots, D,$$

where $X = (x_1, x_2, \ldots, x_n)^T$ is called decision vector, $X \in \Theta \subset R^n$, Θ is search space. $y \in Y$ is objective vector and Y is objective space. g_i, $i = 1, 2, \ldots, D$ is a constraint.

In multi-objective optimization case, the following basic concepts are often used.

Definition 1. Let a decision vector $X_1 \in \Theta$.

1. X_1 is said to dominate a decision vector $X_2 \in \Theta (X_1 \prec X_2)$ if and only if $f_i(X_1) \leqslant f_i(X_2)$ $i = 1, 2, \ldots, k$, and $\exists i \in \{1, 2, \ldots, k\}$ $s.d$ $f_i(X_1) < f_i(X_2)$.
2. X_1 is said to be Pareto optimal if and only if $\neg \exists X_2 \in \Theta$ $s.d$ $X_2 \prec X_1$.
3. $P_S = \{X_1 \in \Theta | \neg \exists X_2 \in \Theta \; s.d \; X_2 \prec X_1\}$ is said to be Pareto optimal set of all Pareto optimal decision vectors.
4. $P_F = \{f(X) = (f_1(X), f_2(X), \ldots, f_M(X)) | X \in P_S\}$ is said to be Pareto optimal front of all objective function values corresponding to the decision vectors in P_S.

Pareto optimal decision vector cannot be improved in any objectives without causing degradation in at least one other objective. When a decision vector is non-dominated on the whole search space, it is Pareto optimal.

3. Problem definition and mathematical modeling

In classical FSSP [21], we have n jobs and m machines. Every job consists of m operations that have to be processed in a specified sequence; All of the jobs must visit machines in the same processing route, starting from machine 1 until finishing on machine m. The operation of job i on machine j lasts for a predetermined amount of time, denoted by $t_{i,j}$. Other assumptions commonly characterized for FSSP are as follows.

- Each machine can process only one operation at a time and each job can be processed by one machine at a time.
- A job can visit a machine once and only once.