

International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France

## A self-adaptive hybrid algorithm for solving flexible job-shop problem with sequence dependent setup time

Ameni Azzouz<sup>a,\*</sup>, Meriem Ennigrou<sup>a</sup>, Lamjed Ben Said<sup>a</sup>

<sup>a</sup> Université de Tunis, Institut Supérieur de Gestion, SMART Laboratory

---

### Abstract

The flexible job shop problem (FJSP) has an important significance in both fields of production management and combinatorial optimization. This problem covers two main difficulties, namely, machine assignment problem and operation sequencing problem. To reflect as close as possible the reality of this problem, the sequence dependent setup time is taken into consideration. For solving such a complex problem, we propose a hybrid algorithm based on a genetic algorithm (GA) combined with iterated local search (ILS). It is well known that the performance of an algorithm is heavily dependent on the setting of control parameters. For that, our algorithm uses a self-adaptive strategy based on : (1) the current specificity of the search space, (2) the preceding results of already applied algorithms (GA and ILS) and (3) their associated parameter settings. We adopt this strategy in order to detect the next promising search direction and maintain the balance between exploration and exploitation. Computational results show that our algorithm provides better solutions than other well known algorithms.

© 2017 The Authors. Published by Elsevier B.V.  
Peer-review under responsibility of KES International

**Keywords:** Flexible job-shop problem, sequence dependent setup times, Hybrid algorithms, Self-adaptive algorithms ;

---

### 1. Introduction

The Flexible Job Shop problem (FJSP), first introduced by<sup>1</sup>, is among the hardest combinatorial optimization problems<sup>2</sup>. FJSP consists of a set of  $n$  jobs that must be processed on a set of  $m$  specified machines. Each job is composed of a specific set of operations, which have to be processed according to a given order. Each operation can be processed by a set of resources and has a processing time depending on the resource used. Recently, many researches have been made to find the near optimal solution of FJSP using a varied range of tools and techniques such as<sup>3,4,5,6</sup> and<sup>7</sup>.

Most scheduling researches reported in the literature ignore the setup times or consider them as a part of the processing time. However, in many real-life situations such as chemical, printing, pharmaceutical and automobile manufacturing<sup>8</sup>, the setup times are not only often required between jobs but they are also strongly dependent on the job itself (independent sequence) and on the previous job that ran on the same machine (sequence dependent). Hence, reducing setup times is an important task for an efficient shop performance. To this end, we consider in this paper the FJSP with sequence dependent setup times (SDST).

---

\* Corresponding author. Tel.: +216-20-357-390  
E-mail address: [ameni.azzouz@isg.rnu.tn](mailto:ameni.azzouz@isg.rnu.tn)

Other existing studies in the literature that take into account the SDST constraint are mainly the work of<sup>9</sup>. This latter presents a Tabu Search (TS) algorithm to solve SDST-FJSP optimizing the makespan criterion. Bagheri and Zandieh<sup>10</sup> propose a variable neighborhood search (VNS) based on an integrated approach to minimize an aggregate objective function. Moreover, Mousakhani formulates the SDST-FJSP as a mixed integer linear programming model to minimize the total tardiness parameter<sup>11</sup>. They applied a metaheuristic based on an iterated local search for the same problem definition. Oddi et al.,<sup>12</sup> consider the SDST-FJSP to minimize the makespan using the iterative flattening search (IFS). González<sup>13</sup> develops a memetic algorithm to minimize the makespan when the tabu search was applied to every chromosome generated by the genetic algorithm. In order to evaluate their model, they used the same benchmark as in<sup>12</sup> and prove that the memetic algorithm has obtained a better result against the IFS. The most recent comprehensive survey of the scheduling problem with setup times is given by<sup>14</sup>.

Most modern algorithms proposed to solve this kind of problem consider the combination of multiple algorithms. This kind of algorithm gives the ability to control the balance between exploration and exploitation of the search space. The practice of combining multiple algorithms continues to attract the interests of researchers in order to exploit the benefits of those algorithms. It is well known that the performance of a hybrid algorithm is heavily dependent on the setting of control parameters. For that, there is an increasing recent trend to consider adaptive mechanisms to alter operator choices and/or their parameters such as an adaptive local search for continuous dynamic optimization problems<sup>15</sup>,<sup>16</sup> and<sup>17</sup>. This latter proposes a single-solution based metaheuristic that perturbs the variables separately in order to select the next search direction. Leon and Xioug<sup>18</sup> propose a differential evolution algorithm based on greedy adaptation of control parameters. Voglis et al.,<sup>19</sup> consider an adaptive memetic algorithm based on particular swarm optimization with variable local search pool size. For that, the authors propose an adaptive selection strategy based on local search improvement scores. Kafafy et al. propose a hybrid evolutionary approach with search strategy adaptation for the continuous domains<sup>20</sup>. This latter uses multiple search strategies such as genetic operators, differential evolution (DE), particle swarm optimization (PSO) and guided mutation operator. In the initial step, all strategies have the same chance to be selected. Next, the more successfully strategy behaved in the previous iteration, the more probability it will be chosen in the current iteration for generating the new solutions. Van Rijn et al.,<sup>21</sup> propose a self adaptive genetic algorithm to solve the container loading problem. Here, the authors adopt only a variable mutation rate using SA3 strategy<sup>22</sup>. In this paper, we investigate the flexible Job-shop problem with sequence-dependent setup times (SDST-FJSP). To solve a such complex problem, we propose a new self-adaptive hybrid algorithm (SAHA) integrating genetic algorithm (GA) and iterated local search (ILS). Our algorithm uses a self-adaptive strategy based on the current specificity of the search space, the preceding results of already applied algorithm (GA and ILS) and their associated parameter settings, in order to select the next search direction and maintain the balance between exploration and exploitation.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation. The proposed SAHA is presented in section 3. Section 4 shows and discuss the performance of SAEA on a set of benchmark problems. Conclusions and some future works are presented in section 5.

## 2. Problem Definition

The problem concerns performing performing  $n$  jobs on  $m$  machines. The set of machines is noted  $M$ ,  $M = \{M_1, \dots, M_k\}$ . Each job  $i$  consists of a sequence of  $n_i$  operations (routing). Each routing has to be performed to complete a job. The execution of each operation  $j$  of a job  $i$  (noted  $O_{ij}$ ) requires one machine out of a set of given machines  $M_{i,j}$  (i.e.  $M_{i,j}$  is the set of machines available to execute  $O_{ij}$ ). The problem is to define a sequence of operations together with assignment of start times and machines for each operation. Assumptions considered in this paper are the following:

- jobs are independent of each other;
- machines are independent of each other;
- one machine can process at most one operation at a time;
- no preemption is allowed;
- all jobs are available at time zero;
- Setup times are dependent on the sequence of jobs. When one of the operations of a job  $t$  is processed before one of those of job  $i$  ( $t \neq i$ ) on machine  $M_k$ , the sequence dependent setup time is  $S_{t,i,k} > 0$ .

Download English Version:

<https://daneshyari.com/en/article/4960624>

Download Persian Version:

<https://daneshyari.com/article/4960624>

[Daneshyari.com](https://daneshyari.com)