



HAL
open science

The Flexible Job Shop Scheduling Problem with Setups and Operator Skills: an application in the textile industry

Tom Perroux, Taha Arbaoui, Leila Merghem-Boulahia

► To cite this version:

Tom Perroux, Taha Arbaoui, Leila Merghem-Boulahia. The Flexible Job Shop Scheduling Problem with Setups and Operator Skills: an application in the textile industry. 18th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2024), Aug 2024, Vienne, Austria. pp.1090 - 1095, <10.1016/j.ifacol.2024.09.125>. <hal-04811448>

HAL Id: hal-04811448

<https://hal.science/hal-04811448v1>

Submitted on 9 Dec 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY-NC-ND 4.0 - Attribution - Non-commercial use - No Derivative Works - International License

The Flexible Job Shop Scheduling Problem with Setups and Operator Skills: an application in the textile industry

Tom Perroux^{*,**} Taha Arbaoui^{**} Leila Merghem-Boulahia^{***}

^{*} EMO SAS, 51 Rue Courtalon, 10000 Troyes, France, (e-mail: tom.perroux@gmail.com)

^{**} INSA Lyon, Université Lumière Lyon 2, Université Claude Bernard Lyon 1, Université Jean Monnet Saint-Etienne, DISP UR4570, Villeurbanne, 69621, France, (e-mail: taha.arbaoui@insa-lyon.fr)

^{***} Laboratory of Computer Science and Digital Society, University of Technology of Troyes, 12 Rue Marie Curie, 10000, Troyes, France, (e-mail: leila.merghem@utt.fr)

Abstract: The Flexible Job Shop Scheduling Problem (FJSSP) is one of the most studied problems in the literature thanks to its practical relevance. Many industrial production scheduling problems can be seen as a FJSSP with different constraints and objectives. The studied problem arises in a textile factory, precisely in the sewing process. Multiple real-world constraints are tackled. Individual skills of operators and different types of setup times are considered: machine-change setup time for operators, colour-change setup times and configuration-change setup times. The objective is to minimize the total tardiness. To the best of the authors' knowledge, this is the first study that tackles this variant of the problem. This paper presents a preliminary study to validate the approach and evaluate the tractability of the industrial problem. A mixed-integer linear programming (MILP) model is proposed and solved using a commercial solver. Real-world data are used to generate more than 1000 problems that are clustered in 4 groups based on the number of jobs to be scheduled, the number of operators and flexibility of their skills. The results show that the proposed MILP model can reach optimality for the small size instances but it is intractable for most large-sized instances. Moreover, the results highlight how diversifying the operators' skills can help find better solutions.

Copyright © 2024 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: Flexible Job Shop Scheduling, Mixed Integer Programming, Tardiness Minimization, Real-World Problem, Case Study

1. INTRODUCTION

In the context of Industry 4.0, textile manufacturers have to adapt their production systems to the varied and rapidly changing demands of their clients. These changes have an impact on the various stages of the textile manufacturing processes from the production of yarn to the distribution of final products to customer. We study the case of an industrial partner, whose factory is located in France. In a textile factory, three main processes appear once the yarn is produced from natural or synthetic fibers (Berthier et al., 2022). First is the knitting process to create fabric rolls from the yarn. Second is the dyeing and treatment process of the fabrics to obtain the desired textile colour and quality. The last is the assembling process to produce the garments.

The studied factory covers the various stages of the assembling process: the cutting process, the sewing process and the preparation process before shipping. The sewing process is the most critical stage in the studied company. The cutting process right before and the preparation process right after are composed of a few operators that

perform one single operation. Additionally, the garments going through the preparation process directly depends on the garments coming from the sewing process. For these reasons, this study focus on the sewing process. The goal is to automate the scheduling of operations going through the sewing process. This stage is typically a Flexible Job Shop Scheduling Problem (FJSSP) with additional setup and operator constraints. New operations are arriving on a daily basis thus the scheduling of the sewing process is at an operational level of decision. Though, client orders have delivery date, thus the quality of the operational decisions can have an impact on the medium to long term satisfaction of client orders. Integrated approaches such as the master scheduling problem have been proposed for the textile industry (Karacapilidis and Pappis, 1996), but the complexity of the sewing process observed in this study make it not suitable. The objective of this study is to propose a decision-making tool that is practical for our industrial partner and that may be later embedded in a larger decision-support system considering multiple stages of the assembling process.

There has been a recent interest for scheduling problems applied to the textile industry. Berthier et al. (2022) proposed a mathematical programming approach to solve the unrelated parallel machine scheduling problem subject to dependent setup times in a textile industry. Mercier-Aubin et al. (2020) applied constraint programming to solve a large Resource-Constrained Project Scheduling Problem with sequence-dependant setup times from a textile industry while aiming at minimizing the weighted tardiness. Ortíz-Barrios et al. (2018) used a Fuzzy Analytic Hierarchy Process and VIKOR methods while considering setup times and transfer batches in a textile industry under a flexible job shop context to minimize the tardiness. Bitar et al. (2021) studied a similar problem and compared three different criteria that are relevant to many industrial cases.

In industrial factories, and especially in flexible contexts, machines are multipurpose and may require a setup to change the configuration before processing a specific operation. Mahmoodjanloo et al. (2020) studied the FJSSP with reconfigurable machine tools modeled as configuration-dependant setup times. They proposed a position-based and a sequenced-based MILP and concluded that the sequenced-based was able to obtain optimal results on small size instances. Moreover, it significantly outperforms the position-based model. Additionally, they proposed a self-adaptive differential evolution algorithm to address larger instances.

Additionally, in the context of the industry 4.0, the number of autonomous tasks is ever growing. Human operators tends to either perform tasks that are too complex to be automated or else to setup and configure the machines to reduce the idle time and increase their efficiency. Therefore, human operators are becoming more versatile and skilled. One has to consider human operator's skills in order to take the best planning decisions. Problems constrained by machine and operator resources may also be referred to as Dual-Resource-Constrained (DRC). Andrade-Pineda et al. (2020) proposed a heuristic for the dual-resource FJSSP in a multi-objective context aiming at minimizing the makespan and due date-related criteria. Gong et al. (2020) introduced a mathematical model for the FJSSP with worker flexibility. They proposed instances for the studied problem and solved the benchmark with a hybrid artificial bee colony algorithm and concluded that their approach is effective on large-scale instances. Berti et al. (2021) pushed further the consideration of human operator with a dual-resource constrained job shop scheduling problem. They studied different models to integrate rest allowance, the impact of rest allowance on the makespan and measured the impact of rest allowance on the system's performance while considering the age and health of the operators in the working environment. Vital-Soto et al. (2023) proposed a non-dominated sorting genetic algorithm (NSGA-II) to solve a multi-objective dual-resource FJSSP with sequence flexibility, such that precedence constraints are expressed with a directed acyclic graph rather than a linear sequence. They aim at minimizing the makespan, weighted tardiness and the maximal worker workload. Shi et al. (2023) studied a dual-resource FJSSP while considering operator boredom. They state that reducing boredom can have a positive effect on the efficiency diminution of operator over time and absenteeism.

To the best of author's knowledge, very few studies considered setup times for DRC-FJSSP. Kress et al. (2019) considered machine operator qualifications and operator-dependent processing times in a FJSSP with sequence-dependent setup times. They aimed at minimizing the makespan and the total tardiness. Wu et al. (2021) studied a DRC-FJSSP subject to setup times due to loading and unloading between operations. They propose both a similarity-based scheduling algorithm and a NSGA-II to minimize makespan and the total setup time. Tan et al. (2021) considered the fatigue of operators in a DRC-FJSSP. Transportation times occur and contribute to the fatigue of operators. They minimize the maximum worker fatigue and makespan with NSGA-II. For a deeper literature analysis on FJSSP, we refer the reader to the review of Dauzère-Pérès et al. (2023).

The remainder of this paper is as follows. The following section will introduce the industrial context and the studied problem. The industrial process of the partner company is described. The subsequent scheduling problem is then detailed with the constraints and the objective function. In Section 3, the proposed MILP model is introduced. Section 4 is devoted to describing the proposed benchmark and the results of the proposed MILP model. The instance groups and parameters are presented. Moreover, the impact of the skill diversity in the operator set is studied and analysed. Finally, the conclusion is to be found in Section 5.

2. INDUSTRIAL CONTEXT AND PROBLEM DESCRIPTION

This study presents the industrial case encountered while automating the planning of the sewing shop. The objective for the company is to satisfy the delivery dates of clients orders. A client order describes the list of garments that have to be delivered before a specific due date. Each order is composed of a list of models and for each model the quantity per size and colour. The company covers all steps of the production process until the finished garments are delivered to the client. The first step in the production is to define the clothing designs of the garments. The designs will be used to define the manufacturing sequences describing the list of operations to follow as well as the fabrics and supplies required for garment making. The fabrics are made in another plant that handle the knitting and dyeing.

Once the required fabrics and furniture are received, cut orders are created following the client's order instructions. A cut order describes the size and the number of articles that must be cut from fabric rolls. Once a cut order is completed, the cut fabrics are grouped in lots of various article quantities. The garments lots can then be scheduled in the sewing process.

In the sewing process, each garment must follow a sequence of operations given by a manufacturing sequence. The manufacturing sequence depends on the nature of the garment, the type of fabric, the order's criteria, the size of the model and the colour of the garment. It describes, for each operation of the model, the machine on which it must be processed and the estimated duration per article. In a lot, each article has the same manufacturing sequence.

Thus, the processing time of an operation on a lot is proportional to the number of articles in the lot. Supplies that are required for the sewing process are added to the lots before the start of the process.

Because of the large variety of operations, articles have to go through different types of machines. Lots are transported between the different machines of the sequence with a trolley. Additionally, each machine may have different configurations and a setup time between two consecutive operations on the same machine occur if the required configurations are different.

For each operation on a machine, an operator is required to perform the operation. Each operator has their own set of skills that defines machines on which they can operate such that they can only process a subset of operations. Most operators can work on multiple machine types. A machine change setup time is required between two consecutive operations processed by the same operator on different machines. Machines having the same type are identical, thus the processing time of an operation does not depend on the machine on which it is scheduled. However, each operator has a personal efficiency that modifies the estimated processing time.

Lastly, each model is associated with a colour that corresponds to the thread colour used through the sewing process. If two operations having different thread colour are processed consecutively on the same machine, a thread change setup is required. This last setup takes only a couple minutes but it shouldn't be neglected as it can occur frequently. The scheduling of the sewing process is thus the assignment of operations to a couple of machine and operator such that one or multiple criteria are optimized. As the company expressed a wish to meet the due dates of the different orders as much as possible, the total job tardiness minimisation is selected to evaluate the solutions.

3. THE PROPOSED MILP MODEL

The problem introduced is modeled as a Flexible job-Shop Scheduling Problem subject to sequence-dependant setup times. Notations used in the MILP for variables and parameters are introduced in Table 1.

A set of jobs I each having a set of operations O_i for each job $i \in I$ are considered with precedence constraints between successive operations. The set of jobs and operations are given by the list of clients orders and their associated manufacturing sequence. The set of compatible machines A_{ij} for an operation o_{ij} is computed from the machine inventory and the machine type and configuration required for operation o_{ij} . The set of qualified operators B_{ij} for an operation o_{ij} is derived based on an operator qualification matrix and the minimum skill level indicated for operation o_{ij} . The processing time p_{ijh} per operation o_{ij} and for each qualified operator $h \in B_{ij}$ is computed based on the estimated processing time per article, the quantity and the efficiency of the operator h . Machine setup times are considered between pairs of operations based on the thread color and machine configuration required (0 if identical). Similarly, a machine change setup is computed between pairs of operations based on the machine type of operations (0 if identical).

| |
|---|
| $I = \{1, \dots, n\}$: set of jobs |
| $O_i = \{o_{i1}, \dots, o_{in_i}\}$: set of operations of job i |
| K : set of machines |
| H : set of operators |
| d_i : due date of job i |
| p_{ijh} : processing time of operation o_{ij} performed by operator h |
| A_{ij} : set of compatible machines for operation o_{ij} |
| B_{ij} : set of qualified operators for operation o_{ij} |
| $s1'_{il}$: thread change setup time between successive jobs i and l on the same machine |
| $s1''_{ijlg}$: configuration change setup time between successive operations o_{ij} and o_{lg} on the same machine |
| $s1_{ijlg} = s1'_{il} + s1''_{ijlg}$: aggregated setup time between successive operations on the same machine |
| $s2_{ijlg}$: machine change setup time between successive operations o_{ij} and o_{lg} processed by the same operator |
| $x_{ijkh} = \begin{cases} 1 & \text{if operation } o_{ij} \text{ is assigned to machine } k \text{ and is performed by operator } h \\ 0 & \text{otherwise} \end{cases}$ |
| $u_{ijlg} = \begin{cases} 1 & \text{if operation } o_{ij} \text{ is performed before operation } o_{lg} \text{ and are performed either on the same machine or by the same operator (or both)} \\ 0 & \text{otherwise} \end{cases}$ |
| $y_{ijlg} = \begin{cases} 1 & \text{if operation } o_{ij} \text{ and operation } o_{lg} \text{ are performed on the same machine} \\ 0 & \text{otherwise} \end{cases}$ |
| $z_{ijlg} = \begin{cases} 1 & \text{if operation } o_{ij} \text{ and operation } o_{lg} \text{ are performed by the same operator} \\ 0 & \text{otherwise} \end{cases}$ |
| $c_{ij} \geq 0$, completion time of operation o_{ij} |
| $p_{ij} \geq 0$, processing time of operation o_{ij} once it is assigned to an operator |

Table 1. Notations for the MILP formulation.

$$\min \sum_{i \in I} \max(c_{in_i} - d_i, 0) \quad (1)$$

$$\sum_{k \in A_{ij}} \sum_{h \in B_{ij}} x_{ijkh} = 1, \forall i \in I, j \in O_i \quad (2)$$

$$p_{ij} = \sum_{h \in B_{ij}} p_{ijh} \sum_{k \in A_{ij}} x_{ijkh}, \forall i \in I, j \in O_i \quad (3)$$

$$c_{ij} \geq c_{ij-1} + p_{ij}, \forall i \in I, \forall j = 2, \dots, n_i \quad (4)$$

$$c_{i0} \geq p_{i0}, \forall i \in I \quad (5)$$

$$c_{lg} \geq c_{ij} + p_{ij} + s1_{ijlg} y_{ijlg} + s2_{ijlg} z_{ijlg} - M(1 - u_{ijlg}), \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, j \neq g \quad (6)$$

$$y_{ijlg} \geq \sum_{h \in B_{ij}} x_{ijkh} + \sum_{h \in B_{lg}} x_{lghh} - 1, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, h \neq g, k \in A_{ij} \cap A_{lg} \quad (7)$$

$$z_{ijlg} \geq \sum_{k \in A_{ij}} x_{ijkh} + \sum_{k \in A_{lg}} x_{lghh} - 1, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, h \neq g, h \in B_{ij} \cap B_{lg} \quad (8)$$

$$2(u_{ijlg} + u_{lgi}) \geq y_{ijlg} + z_{ijlg}, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, j \neq g \quad (9)$$

$$x_{ijkh} \in \{0, 1\}, \forall i \in I, j \in O_i, k \in A_{ij}, h \in B_{ij} \quad (10)$$

$$u_{ijlg} \in \{0, 1\}, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, h \neq g \quad (11)$$

$$y_{ijlg} \in \{0, 1\}, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, h \neq g \tag{12}$$

$$z_{ijlg} \in \{0, 1\}, \forall i \in I, j \in O_i, l \in I, g \in O_l, i \neq l, h \neq g \tag{13}$$

The objective and constraints for the MILP problem are presented hereafter. The objective function (1) is to minimize the total tardiness of jobs. Constraint (2) ensures that there is exactly one operator and one machine assigned to an operation. Constraint (3) links the effective processing time p_{ij} of operation o_{ij} to the operator performing the operation. The order of the job operation sequence is ensured by Constraint (4). Constraint (5) forbids operations to start before the start of the planning horizon. Constraint (6) prevents overlap between consecutive operations that are either processed by the same operator or on the same machine, furthermore the setup times are added (setup times may be 0 if there is no setup required between the consecutive operations). Setup times are treated as minimal distance between operations. The completion time of subsequent jobs are bound through precedence constraints. The value of variables y_{ijlg} is set through Constraint (7) such that if operations o_{ij} and o_{lg} are scheduled on the same machine y_{ijlg} is equal to 1. Similarly, Constraint (8) are defined to determine the value of variables z_{ijlg} if o_{ij} and o_{lg} are performed by the same operator. Finally, the value of variables u_{ijlg} or u_{lgij} is forced to 1 by Constraint (9) if o_{ij} and o_{lg} are either scheduled on the same machine or operator (or both), such that operations o_{ij} and o_{lg} cannot overlap. Constraints (10)-(13) are the integrity constraints. The performance of the MIPL model are evaluated in the next section.

4. COMPUTATIONAL RESULTS

In this section the instance generation procedure is first presented. Then the results of the computational experiment are analyzed to evaluate the performance of the MILP approach.

4.1 Instance Generation Procedure

The approach is evaluated on a collection of instances generated based on the data collected from our industrial partner. The industrial case is composed of a set of 92 machines and 30 operators. There are 25 different types of machines and machines may have several possible configurations. Operations require a specific couple of machine type and configuration. Each operator has a set of skills from which the set of operations they can perform is deduced. Moreover, each operator has a personal efficiency that impact the effective time spent to perform an operation.

The production is organised in groups of operators, each group working on a subset of client orders. Some operators always work in the same group while others may change depending on their skill and the workload of the groups. Some groups are dedicated to only one client, thus there is a low variety of operations. On the other hand, some groups deal with multiple clients at the same time, thus there is much more operations types and machines to consider.

Table 2. Description of skill density

| Group | low | avg | high |
|-------|-----|-----|------|
| 0 | 42% | 50% | 75% |
| 1 | 38% | 50% | 75% |
| 2 | 24% | 33% | 50% |
| 3 | 28% | 33% | 50% |

Table 3. Description of instances

| Group | min | med | max | n | m | Op. |
|-------|-----|-----|-----|------------|------------|---------------|
| 0 | 6 | 8 | 9 | 1-11 (4) | 7-24 (13) | 6-129 (43) |
| 1 | 4 | 6 | 8 | 3-27 (12) | 4-22 (11) | 12-115 (44) |
| 2 | 10 | 14 | 17 | 6-31 (16) | 14-42 (29) | 22-154 (87) |
| 3 | 9 | 15 | 20 | 33-94 (54) | 25-85 (55) | 197-697 (367) |

The instance generation procedure is as follows. First multiple workforce configurations are generated. Based on historical data, the sets of permanent operators and optional operators are determined: respectively operators that are always working in this group and operators that work in this group occasionally. Then, for each operator $h \in H$, their minimal set of skills collected from the company data is denoted S_{low}^h . We determine two additional sets of skills, respectively average and high, S_{avg}^h and S_{high}^h by adding skills to operator h such that $S_{low}^h \subset S_{avg}^h \subset S_{high}^h$. Skills are added randomly until the target skill density is met. The description of skill density per group is given in Table 2. By adding skills, the company wanted to know whether training operators on other skills can benefit the flexibility of the sewing shop.

Instances of various sizes are generated by considering different production groups. Instances from production groups 0 and 1 are considered as the small size instances due to the low number of operators in the group and the number of operations being the smallest on average. Instances from production group 2 correspond to medium size instances. On average, this group is twice as large as small size instances. Lastly, group 3 represent the large size instances. The number of operators is similar to the medium size instances but the number of operations and machines is much higher.

The data associated to client orders were collected each week for 28 weeks. For each week one instance is generated for each combination (36 in total) of instance parameters : group in $\{0, 1, 2, 3\}$, group size in $\{min, med, max\}$, and skill density in $\{low, avg, high\}$. Instance characteristics per group are presented in Table 3. For each group, the number of operators in the minimal group, the medium group and the maximal group (respectively min, med and max) are presented. Moreover, the range and average (in parenthesis) are shown for the number of jobs (n), the number of machines (m) and the number of operations (Op.) over the 28 weeks.

4.2 Computational Results

The collection of instances was solved with Cplex on a Intel Xeon Gold 6128 3.40GHz CPU. The timeout was set to 3600 seconds. The results are analyzed in the remainder of the section.

Table 4. Results comparison per group, group size and skill density

| Group | gsize | density | O | F | N | Time |
|-------|-------|---------|----|----|----|---------|
| 0 | min | low | 22 | 4 | 2 | 1112.99 |
| 0 | min | avg | 21 | 4 | 3 | 1377.43 |
| 0 | min | high | 21 | 3 | 4 | 1256.09 |
| 0 | med | low | 22 | 2 | 4 | 1110.45 |
| 0 | med | avg | 21 | 2 | 5 | 1321.32 |
| 0 | med | high | 20 | 3 | 5 | 1397.73 |
| 0 | max | low | 20 | 4 | 4 | 1401.35 |
| 0 | max | avg | 21 | 3 | 4 | 1198.64 |
| 0 | max | high | 21 | 2 | 5 | 1208.82 |
| 1 | min | low | 16 | 12 | 0 | 2290.21 |
| 1 | min | avg | 14 | 14 | 0 | 2592.87 |
| 1 | min | high | 10 | 18 | 0 | 2733.93 |
| 1 | med | low | 14 | 14 | 0 | 2576.74 |
| 1 | med | avg | 13 | 14 | 1 | 2636.33 |
| 1 | med | high | 10 | 16 | 2 | 2646.14 |
| 1 | max | low | 13 | 15 | 0 | 2433.86 |
| 1 | max | avg | 15 | 12 | 1 | 2264.2 |
| 1 | max | high | 12 | 13 | 3 | 2558.8 |
| 2 | min | low | 10 | 13 | 5 | 2854.13 |
| 2 | min | avg | 8 | 10 | 10 | 2882.62 |
| 2 | min | high | 7 | 7 | 14 | 3191.73 |
| 2 | med | low | 8 | 14 | 6 | 2864.75 |
| 2 | med | avg | 8 | 7 | 13 | 2945.86 |
| 2 | med | high | 7 | 6 | 15 | 3062.02 |
| 2 | max | low | 7 | 14 | 7 | 2945.29 |
| 2 | max | avg | 6 | 11 | 11 | 3044.59 |
| 2 | max | high | 5 | 7 | 16 | 3188.98 |
| 3 | min | low | 0 | 1 | 27 | 3608.26 |
| 3 | min | avg | 0 | 0 | 28 | 3615.9 |
| 3 | min | high | 0 | 0 | 28 | 3605.43 |
| 3 | med | low | 0 | 1 | 27 | 3664.72 |
| 3 | med | avg | 0 | 0 | 28 | 3608.76 |
| 3 | med | high | 0 | 0 | 28 | 3397.51 |
| 3 | max | low | 0 | 0 | 28 | 3624.3 |
| 3 | max | avg | 0 | 0 | 28 | 3624.57 |
| 3 | max | high | 0 | 0 | 28 | 3269.69 |

Table 4 presents the results obtained per group, group size and skill density over the 28 weeks. We present the number of instances for which we obtained respectively an optimal solution (O), a non-optimal feasible solution, (F) and no solution before the timeout (N). Time is the mean time spent solving. Results reveal that the model was able to obtain an optimal solution for most small size instances (group 0 and 1). No feasible solution was obtained on half of the medium size instances (group 2) but optimality was reached on a quarter of instances. As for large size instances the model could hardly obtain a feasible solution. It reveals that our approach is suitable to solve instances on small production groups. It could be used on medium size production groups but one should be cautious about the solution quality. Our approach is not suitable for industrial size instances on large production groups.

To better understand the impact of group size and skill density, we present Table 5 for which results of the same group have been sum up respectively per group size and skill density. The impact of group size and skill density is evaluated hereafter: The number of instances solved with state O, F and N is similar over the various group sizes and skill densities. Nevertheless, one can observe that the

Table 5. Aggregated results comparison per group size and skill density

| Group | gsize | O | F | N | density | O | F | N |
|-------|-------|----|----|----|---------|----|----|----|
| 0 | min | 64 | 11 | 9 | low | 64 | 10 | 10 |
| 0 | med | 63 | 7 | 14 | avg | 63 | 9 | 12 |
| 0 | max | 62 | 9 | 13 | high | 62 | 8 | 14 |
| 1 | min | 40 | 44 | 0 | low | 43 | 41 | 0 |
| 1 | med | 37 | 40 | 3 | avg | 42 | 40 | 2 |
| 1 | max | 40 | 40 | 4 | high | 32 | 47 | 5 |
| 2 | min | 25 | 30 | 29 | low | 25 | 41 | 18 |
| 2 | med | 23 | 27 | 34 | avg | 22 | 28 | 34 |
| 2 | max | 18 | 32 | 34 | high | 19 | 20 | 45 |
| 3 | min | 0 | 1 | 83 | low | 0 | 2 | 82 |
| 3 | med | 0 | 1 | 83 | avg | 0 | 0 | 84 |
| 3 | max | 0 | 0 | 84 | high | 0 | 0 | 84 |

Table 6. Objective comparison per group size and skill density

| Group | Week | min | | | med | | | max | | |
|-------|------|-------|------|------|------|-----|------|------|-----|------|
| | | low | avg | high | low | avg | high | low | avg | high |
| 0 | 0 | 3927 | 76% | 56% | 76% | 68% | 54% | 59% | 59% | 50% |
| 0 | 1 | 7029 | 67% | 50% | 67% | 61% | 49% | 57% | 57% | 45% |
| 0 | 2 | 5754 | 95% | 86% | 95% | 95% | 85% | 95% | 87% | 81% |
| 0 | 3 | 2394 | 57% | 39% | 58% | 52% | 38% | 52% | 40% | 33% |
| 0 | 7 | 5146 | 43% | 6% | 30% | 30% | 6% | 23% | 23% | 6% |
| 0 | 8 | 166 | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 0 | 9 | 483 | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 0 | 10 | 1380 | 0% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 0 | 11 | 3180 | 39% | 0% | 25% | 25% | 0% | 22% | 22% | 0% |
| 0 | 12 | 5233 | 50% | 31% | 51% | 49% | 31% | 43% | 43% | 24% |
| 1 | 1 | 1437 | 89% | 61% | 94% | 62% | 59% | 90% | 60% | 52% |
| 1 | 7 | 14530 | 83% | 84% | 98% | 83% | 83% | 96% | 83% | 80% |
| 1 | 8 | 21661 | 95% | 85% | 97% | 86% | 84% | 95% | 84% | 81% |
| 1 | 10 | 426 | 0% | 0% | 21% | 0% | 0% | 0% | 0% | 0% |
| 1 | 11 | 483 | 11% | 11% | 46% | 0% | 0% | 0% | 0% | 0% |
| 1 | 19 | 565 | 100% | 91% | 100% | 91% | 91% | 100% | 91% | 91% |
| 1 | 20 | 2716 | 100% | 81% | 100% | 81% | 80% | 100% | 75% | 75% |
| 1 | 21 | 5154 | 99% | 86% | 99% | 86% | 86% | 99% | 85% | 83% |
| 1 | 24 | 8276 | 93% | 80% | 96% | 82% | 79% | 93% | 79% | 75% |
| 2 | 8 | 21661 | 98% | 86% | 95% | 90% | 84% | 95% | 84% | 81% |
| 2 | 10 | 258 | 9% | 0% | 0% | 0% | 0% | 0% | 0% | 0% |
| 2 | 11 | 1649 | 23% | 0% | 4% | 0% | 0% | 3% | 0% | 0% |
| 2 | 12 | 2327 | 69% | 49% | 68% | 53% | 49% | 62% | 50% | 49% |

number of instances solved optimally (O) is slightly lower as the group size increases (except for group 1), and also as the density increases. In particular, for group 2 the number of instances solved optimally is quite different depending on the group size. Similarly, the number of unsolved instances (N) is moderately higher as the group size increases (except for group 0), or as the density increases. In particular, for group 2 both the number instances solved optimally and the number of unsolved instances is quite different depending on the skill density. Overall, the impact of group size and skill density on the instance complexity is small for groups 0, 1 and 3 but for group 2 the impact is moderate. Thus the impact is mostly group-dependent.

Additionally, the impact of group size and skill density on the objective function is presented in Table 6. Weeks for which all instances were solved optimally were selected to perform the analysis. The value of the instance with the minimal size and the lowest density is presented, and the relative values are shown in percent for other instances of the same week. For clarity, the relative value can range from 100% if it is unchanged down to 0% if it reaches objective value 0.

Column max with high density reveals that increasing both the number of workers and the skills of workers can have a large impact on the objective value as for most weeks the relative value was lower than 50%. However, for some weeks (7-8-19-20-21) the gain was moderate (more than 80%). Comparing columns min with high density and max with low or avg density shows that having higher skilled workers can have more impact on the objective than having more workers with a lower skill density (group 0 weeks 1-2-3-7, group 1 weeks 7-8-20-21-24, group 2 weeks 8-12). Overall, our results shows that the workforce and the qualifications of workers can significantly impact production outcomes. Nevertheless, one can observe that the results are very week-dependent. For some weeks, reinforcing the workforce does not provide significant benefits and would instead lead to inefficient utilization of human resources. From a strategic management perspective, one have to adapt the workforce to the varying demand of each week in order to optimize the resource allocations. It is also important to analyze the impact of training new skills to workers in order to obtain the most efficient workforce.

5. CONCLUSION

In this paper, the scheduling of operations in a sewing shop that occur in a French textile company was studied. The problem was modeled as a dual-resource flexible job shop scheduling problem subject to various industrial constraints and a MILP model was proposed. A large collection of instances was proposed based on the company's historical data and led to a computational analysis. Results showed that the approach is suitable to solve small size instances but is intractable for the larger industrial size instances. Moreover, operator skills were varied and analyzed and results showed that increasing the skill variety of operators can lead to a higher flexibility of shop allowing the MILP to find better solutions.

As future research, one wants to explore other solution approaches to tackles the larger real-world instances in weeks with the highest number of operations. Recent advances in Constraint Programming has proved very effective for scheduling problems. Furthermore, heuristics methods can produce efficient solutions in a short amount of time are promising to address the largest real-world instances faced in the studied sewing shop.

REFERENCES

- Andrade-Pineda, J.L., Canca, D., Gonzalez-R, P.L., and Calle, M. (2020). Scheduling a dual-resource flexible job shop with makespan and due date-related criteria. *Annals of Operations Research*, 291, 5–35.
- Berthier, A., Yalaoui, A., Chehade, H., Yalaoui, F., Amodeo, L., and Bouillot, C. (2022). Unrelated parallel machines scheduling with dependent setup times in textile industry. *Computers & Industrial Engineering*, 174, 108736.
- Berti, N., Finco, S., Battaià, O., and Delorme, X. (2021). Ageing workforce effects in dual-resource constrained job-shop scheduling. *International Journal of Production Economics*, 237, 108151.
- Bitar, A., Dauzère-Pérès, S., and Yugma, C. (2021). Unrelated parallel machine scheduling with new criteria: Complexity and models. *Computers & Operations Research*, 132, 105291.
- Dauzère-Pérès, S., Ding, J., Shen, L., and Tamssaouet, K. (2023). The flexible job shop scheduling problem: A review. *European Journal of Operational Research*.
- Gong, G., Chiong, R., Deng, Q., and Gong, X. (2020). A hybrid artificial bee colony algorithm for flexible job shop scheduling with worker flexibility. *International Journal of Production Research*, 58(14), 4406–4420.
- Karacapilidis, N.I. and Pappis, C.P. (1996). Production planning and control in textile industry: A case study. *Computers in industry*, 30(2), 127–144.
- Kress, D., Müller, D., and Nossack, J. (2019). A worker constrained flexible job shop scheduling problem with sequence-dependent setup times. *OR Spectrum*, 41(1), 179–217.
- Mahmoodjanloo, M., Tavakkoli-Moghaddam, R., Baboli, A., and Bozorgi-Amiri, A. (2020). Flexible job shop scheduling problem with reconfigurable machine tools: An improved differential evolution algorithm. *Applied Soft Computing*, 94, 106416.
- Mercier-Aubin, A., Gaudreault, J., and Quimper, C.G. (2020). Leveraging constraint scheduling: a case study to the textile industry. In *Integration of Constraint Programming, Artificial Intelligence, and Operations Research: 17th International Conference, CPAIOR 2020, Vienna, Austria, September 21–24, 2020, Proceedings 17*, 334–346. Springer.
- Ortiz-Barrios, M., Neira-Rodado, D., Jiménez-Delgado, G., and Hernández-Palma, H. (2018). Using fahp-viktor for operation selection in the flexible job-shop scheduling problem: A case study in textile industry. In *Advances in Swarm Intelligence: 9th International Conference, ICSI 2018, Shanghai, China, June 17–22, 2018, Proceedings*. Springer.
- Shi, J., Chen, M., Ma, Y., and Qiao, F. (2023). A new boredom-aware dual-resource constrained flexible job shop scheduling problem using a two-stage multi-objective particle swarm optimization algorithm. *Information Sciences*, 643, 119141.
- Tan, W., Yuan, X., Wang, J., and Zhang, X. (2021). A fatigue-conscious dual resource constrained flexible job shop scheduling problem by enhanced nsga-ii: An application from casting workshop. *Computers & Industrial Engineering*, 160, 107557.
- Vital-Soto, A., Baki, M.F., and Azab, A. (2023). A multi-objective mathematical model and evolutionary algorithm for the dual-resource flexible job-shop scheduling problem with sequencing flexibility. *Flexible Services and Manufacturing Journal*, 35(3), 626–668.
- Wu, X., Peng, J., Xiao, X., and Wu, S. (2021). An effective approach for the dual-resource flexible job shop scheduling problem considering loading and unloading. *Journal of Intelligent Manufacturing*, 32, 707–728.