AN EFFECTIVE APPROACH FOR SOLVING A JOB SHOP PROBLEM UNDER COMPONENTS CONSUMPTION AND UNIDIRECTIONAL TRANSPORT CONSTRAINTS: A CASE STUDY

Abdelkader HADRI[1, 3], Hacene SMADI[2], Aimed eddine BOUGLOULA[1] and Fayçal BELKAID[23,4]

[1] Department of Industrial Engineering, University of Batna 02, route de Constantine, Fesdis Batna 05078, Algeria.
E-mail : hadriabdelkader@yahoo.fr , E-mail : Imade_ed@yahoo.fr.
[2] Department of Electrical and Electronic Engineering, University of Tlemcen, PB 230, Tlemcen, 13000, Algeria.
[3] Manufacturing Engineering Laboratory of Tlemcen (MELT), PB 230, Tlemcen, 13000, Algeria,
E-mail : faycal.belkaid@univ-tlemcen.dz
[4] University of Batna 02, Laboratory of Automation and Manufacturing (LAP), Fesdis Batna 05078, Algeria.
E-mail : h.smadi@hotmail.com

***ABSTRACT**: In this work we are interested in the job shop problem with two constraints, availability of components and unidirectional transport of jobs. The study presented here is focused on the flexible iCIM 3000 system which consists of four stations (machines), an AS / RS storage system and a parts transfer system. In this system, the movement of parts is carried out in one direction, which means all the parts follow the same direction of movement. Indeed, to be processed in a machine, each part needs a number of non-renewable resources which are available in limited quantity. Our goal is to find the best parts sequencing which minimizes the total accomplishing time (Makespan) by considering transport of parts and the availability of consumable resources. A several heuristics based on priority rules are proposed and a large-scale experiment was performed in order to analyze the performance of the proposed methods*

***KEYWORDS**: Job shop scheduling; components consumption; transport constraint; heuristics*

1 INTRODUCTION

Managing flexible manufacturing systems is one of the most important challenges for companies in their production management and control systems. In this type of system, the resources are generally numerically controlled by machines that can process several types of parts. These parts are stored and transported by an automated handling system (Zhou & al, 2019). Indeed, scheduling in this type of shop has a direct and major impact on efficiency and production costs. To this fact, it has attracted a lot of attention from researchers since 1956 (Zhang & al, 2019).

However, this problem presents one of the most difficult problems in production management since it belongs to the NP combinatorial optimization problems. His difficulty lies in the fact that we should process a variety of specific products on different types of machines with different sequences (Kassu & Eshetie, 2015).

For several years, a lot of work has been focused on solving the job shop scheduling problem in a classical context with or without constraints. In (Lin, & al, 2010) the authors proposed a new hybrid swarm intelligence algorithm in order to solve the job-shop scheduling problem with the objective to minimize the makespan. For the same objective, Banharnsakun and al applied the Best-so-far ABC method to solve the job-shop scheduling problem (Banharnsakun & al, 2012).

Recently, a series of works have dealt with the job shop problem. In fact, this problem remains to this day among the real concerns of researchers given its crucial importance. For instance, Dabaha and al, presented a new approach based on the Branch-and-Bound method for solving the problem of job shop with blocking (Dabaha & al, 2018). Further, Gong and al. (Gong & al, 2019) proposed an Effective Mimetic Algorithm (EMA) to solve the multi-objective job shop scheduling problem. In (Gong, and al, 2020) a meta-heuristic algorithm has been proposed to solve the job shop problem with no-wait jobs constraint.

In domain of scheduling, the constraint of non-renewable resources (**CRs**) presents a new challenge. In (Tabrizi & al, 2019), the authors check to obtain the best project scheduling that minimizes makespan by considering the existence of renewable and non-renewable resources. A mixed-integer programming model is developed to present simultaneous project scheduling planning and material procurement problems.

The first work in the job shop scheduling field introducing the **CRs** constraint is the work done in (Grabowski & Janiak, 1987). In this work, the processing time for each operation is represented by a function of the required quantity of the non-renewable resource. In order to solve this problem, the authors proposed an algorithm based on the disjunctive graph theory as well as a Branch-and-Bound technique. In the same context, Toker et al. (Toker & al, 1994) developed an approximate algorithm and introduced two lower bounds to solve the problem of *n* job on *m* machine with non-renewable resource requirements.

Introducing the transport constraint in scheduling problems study for flexible production systems is very important and allows the manager to come much closer to reality. However, this consideration makes the scheduling problem more difficult and more complex (Caumond, 2006). Within this framework, several works have been elaborated in order to solve the job shop with the transport problem. The work of Li and al. (Li & al, 2018), are focused on a three machines job shop problem with an intermediate transfer time between these machines. The objective of this work was to find a better solution that minimizes the makespan by proposing a new model and using heuristics to achieve this goal.

The work of Gondran and al. (Gondran & al, 2018) studied a new objective function for the job shop problem with transport constraints. This objective lies in seeking a solution that minimizes the extent of the makespan and then maximizes the quality of service. A model based on linear mixed integer programming (MILP) was proposed by Heger et al (Heger & Voß, 2019) in order to find an optimal solution to the scheduling problem of AGVs in a job shop environment with blocking. Another mathematical programming model was proposed in (Homayouni, & Fontes, 2019) whose aim is to program the problem of joint production and transport scheduling in flexible manufacturing systems.

Priority rules are widely used in solving NP-hard problems in many areas of production and logistics, especially in the environment of job shop floor (Đurasević & Jakobović, 2018). Two new heuristics have been proposed by Rajendran et al (Rajendran & al, 2017) in order to solve the problem of permutation flow-shop scheduling. The performance of these heuristics has been studied and the results obtained reveal that the proposed priority rules are simple and efficient, and improve the solutions for many cases of problems. In (Ozturk & al, 2019), the authors described two new approaches to extract the priority rules for multi-objective dynamic scheduling problems in the job shop workshop using simulation and programming of the expression of genes.

Through this review of the literature, we note that there are few works that focus on the job shop scheduling problems in the presence of transport and availability of components constraints. Our work which is an extension of work presented in (Belkaid & al, 2018), on the other hand, emphasizes this aspect by introducing the objective of respecting a limited number of components in a job shop scheduling problem with unidirectional transport constraints.

In this problem, it is planned to treat a set of jobs (parts) on four machines whose movement of jobs between these machines to be carried out in a single direction that is mean all the parts follow the same direction of movement. To solve this optimization problem, we proposed several heuristic methods that allow finding a better sequencing of jobs to minimize the maximum processing time (Makespan). Several simulations have been studied and the results obtained demonstrate the effectiveness of the proposed approach.

2 PROBLEM DESCRIPTION AND CASE STUDY

The case study is represented by the flexible manufacturing system iCIM 3000 located in the Manufacturing Engineering Laboratory of Tlemcen (MELT), Algeria. The iCIM 3000 is one of the latest relevant solutions proposed by Festo Didactic for the training of students and also to meet the scientific needs of the research centers interested in manufacturing. This system (shown in figure 1), consists of:

- (1): Turning machine with flexible robot feeder is responsible for the production of the single rotary parts,
- (2): Milling machine with flexible robot feeder, on the holes of different diameters can be made,
- (3): Flexible robot Assembly Cells, the function of this station is to assemble products from semi-products created by the CNC machines or other products stocked in the AS/RS station,
- (4): Quality and Handling station, which is responsible for the workpieces testing and the manual feeding of the system with pallets
- (5): The AS (Automatic Storage) / RS (Retrieval System), ensuring the storage

of raw materials, semi-final products as well as final products

- (6): An FMF-Pallet conveyor system, which is used to transfer products from the stock to the stations and /or vice versa.

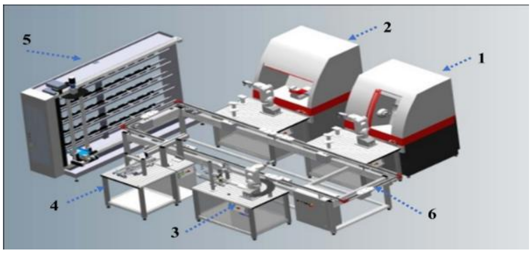

Fig. 1 3D Configuration of the iCIM 3000 system (Sobrino & al, 2013).

The movement of parts between stations is carried out on standard pallets of unit capacity, that is to say, that these pallets can transport any type of parts but with only one part at a time. The pallets are always available and move, as shown in figure 2, continuously on the conveyor in one direction with a constant speed. To simplify the problem, we assume that the number of pallets equals the number of parts to be processed. This assumption makes it possible to avoid the problem of blocking due to the saturation of the conveyor.

Indeed, the processing of parts in Assembly Cell requires the consumption of a number of components. In this station, the final product produced mounted by the iCIM 3000 system is a desk set. This product presented in figure 3, consists of five parts such as base plate, penholder, thermometer device used for temperature measurement, hygrometer device used for humidity measurement.

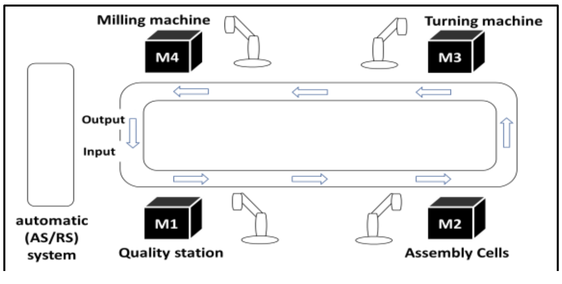

Fig. 2 The movement direction of parts in the iCIM 3000 system

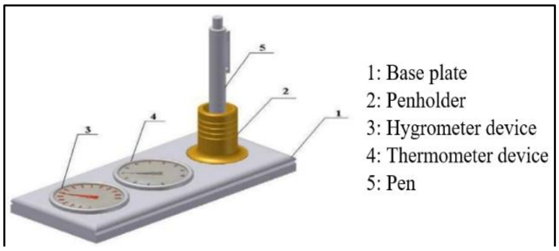

Fig. 3 The iCIM 3000 final product

Depending on the number and type of components to be assembled and the number of holes in the base plate, the system allows the production of several variations of this product.

3 PROBLEM FORMULATION

The problem considered here can be represented as a job shop scheduling problem with constraints of consumable resources and unidirectional transport. This problem consists in scheduling n jobs N={J_1,J_1, ...J_n} on four machines M={M_1, M_1, M_3, M_4}. Each job J_j consists of n_j operations to be carried out in a determined order. Each operation O_j^k of job J_j must be performed on machine M_k without preemption for a processing time P_j^k. For the operation O_j^k to be executed, the job J_j must be transported to the machine M_k with a travel time denoted T_j^k.

Indeed, each job J_j at a given time and before it is processed in the machine M_k, can be either:

- (*i*) processed in a machine $M_{k'}$, (previous machine of M_k in the operating range);
- (*ii*) transported by a pallet;
- (*iii*) in the input buffer (stock) of the M_k machine while awaiting either the availability of the machine or the arrival of the required amount of consumable resources.

The objective is to find the best scheduling that minimizes the total processing time for parts (jobs) by taking into account two constraints; transport constraint and consumable resource constraint. Then, the objective fonction (equation 1) aims to optimize the total execution time C_{max}

Min (C_{max}) = min [$\max_j^k(C_j^k)$] (1)

whose C_j^k determines the accomplishing time of job J_j in M_k which is conditioned by three constraints;

- the availability of the job J_j (including the accomplishing time of its processing on the previous machine and the time it takes to travel),
- the availability of the machine M_k
- and the availability of the consumable resources required to carry out this job.

Therefore the C_j^k is calculated by the following equation:

$$C_j^k = \max\left[\left(C_j^{k'} + T_j^k\right), D_k,\ DRC_j^k\right] + P_j^k \quad (2)$$

Whose $C_j^{k\prime}$ determine the accomplishing time of J_j on the machine $M_{k\prime}$, D_k defines the availability of the machine M_k, $(C_j^{k'} + T_j^k)$ determine the availability of job J_j and DRC_j^k indicates the availability of consumable resources.

The goal is to minimize the time to exit the last job from the system. It is therefore essential to add the time of movement of the parts towards the output of the system at the accomplishing time on the last machine. this means that for the job J_j which ends its execution on the machine M_g (the last machine according to its operating range), the end date C_j^g of this job becomes $C_j^g + T_j^S$.

Typically, the problem takes the following assumptions into consideration:

- Each job has its own routing, which is known before the beginning of the production.
- The processing time is known at the beginning and it includes tool change, set up, and machining times.
- Each machine can process without interruption one job at a time
- The blocking problem is not considered, i.e. each job can be moved to the next machine (without delay) once it is processed in the previous machine.
- Each job can be handled in one machine at a time.
- Job passes in only one direction
- The pallet transports only one job at a time and in one direction
- All machines and jobs are available from time zero.
- Each machine has a stock with unlimited capacity.

4 RESOLUTION APPROACHES

Solving the job shop problem in an optimal way is in most cases very difficult because of its highly combinatorial nature (Zhou & al, 1991). However, The constraint of transport and consumable resources makes the problem more complicated. The exact methods, in this case, require a computational effort that increases exponentially with the size of the problem. So, approximate methods are strongly preferred for solving this type of problem. These methods make it possible to find acceptable solutions within a reasonable time.

In this paper and in order to solve the JSPTRC problem, we have proposed to study the performance of six heuristics based on the priority rules. The objective is to analyze the influence of the transport constraint on the choice of approach which leads to finding the best solutions. In addition, the evaluation criterion that we have chosen to determine the sequencing of jobs in the six proposed heuristics is essentially based on the relationship between processing time and transport time. The choice of this ratio permits to determine, on the one hand, the relation which exists between the time elapsed during the treatment of jobs and the time necessary for their movements, and on the other hand, the influence of the parts transporting time on the sequence of jobs.

To fully explain these methods and our choice concerning the parameters and criteria fixed in each method, we have chosen an illustrative example defined by a problem of four jobs to be executed on the four machines. Table 1 shows the time required to process jobs on the machines as well as the number of resources requested by each of these jobs. The arrival of resources follows a staircase curve that is determined by the arrival of a resource ($Qarr_t^c$=1) every five units of time ($Tarr_t^c$=5).

Table 1. Jobs information's

Jobs	Machines (processing time)	$Qdem_j^c$
J_1	$M_3(2)$ $M_1(3)$ $M_2(5)$ $M_4(25)$	10
J_2	$M_1(15)$ $M_2(10)$ $M_3(20)$ $M_4(8)$	8
J_3	$M_3(7)$ $M_2(14)$ $M_1(5)$ $M_4(5)$	4
J_4	$M_2(9)$ $M_1(4)$ $M_3(3)$	-

Indeed, the resolution of the JSPTRC problem based on this example requires, in a first step, the determination of the travel times of the jobs between stations. So, according to Table 1 which defines the transport time between the different stations and according to the operating ranges of the jobs, we can determine the matrix of times relating to the movements of the jobs during their trajectories in the system. This travel time matrix is presented in table 2.

4.1 Job shop short accumulation processing time (JSSPTcum)

this heuristic consists of ordering jobs in the increasing order of the cumulative processing time of jobs on each machine. For this heuristic, when a machine is released, the job chosen to be processed first is the one with the lowest accumulated value. This accumulated value is calculated by the sum of durations of all previous operations (that is, operations up to the operation to be performed on that machine).

Tab 2. Time required to move jobs between sections

Jobs	Destination		Displacement Time
	Start	Arrivals	
$\boldsymbol{J_1}$	In put	M_3	9.5
	M_3	M_1	8
	M_1	M_2	5
	M_2	M_4	8
	M_4	out put	1.5
$\boldsymbol{J_2}$	In put	M_1	1.5
	M_1	M_2	5
	M_2	M_3	3
	M_3	M_4	5
	M_4	out put	1.5
$\boldsymbol{J_3}$	In put	M_3	9.5
	M_3	M_2	13
	M_2	M_1	11
	M_1	M_4	13
	M_4	out put	1.5
$\boldsymbol{J_4}$	In put	M_2	6.5
	M_2	M_1	11
	M_1	M_3	8
	M_3	out put	6.5

Table 3. Cumulative processing time on machines

Jobs	**Machines**			
	$\mathbf{M_1}$	$\mathbf{M_2}$	$\mathbf{M_3}$	$\mathbf{M_4}$
J_1	5	10	2	35
J_2	15	25	45	53
J_3	26	21	7	31
J_4	13	9	16	-

Table 3 shows the cumulative durations on each machine. According to the heuristic JSSPT, the problem solution is defined by in Table 4. The C_{max} calculated from this sequencing is equal to 168.5 units of time.

Tab 4. Job order according to JSSPTcum

	Machines			
	$\boldsymbol{M_1}$	$\boldsymbol{M_2}$	$\boldsymbol{M_3}$	$\boldsymbol{M_4}$
1	J_1	J_4	J_1	J_3
2	J_4	J_1	J_3	J_1
3	J_2	J_3	J_4	J_2
4	J_3	J_2	J_2	-

4.2 The heuristic JSSPTtrain

For this heuristic, we assume that the sequencing of the jobs on the machines is done according to the increasing order of the ratio of times Rtd_j. This report (described by equation 3) defines a relationship between the amount of time allocated for the movement of a job and the time required to process this job.

$$Rtd_j = \frac{STT_j}{STD_j} = \frac{\sum P_j^k}{\sum T_j^k} \quad (3)$$

In this equation, STT_j determines the sum of the processing times of all operations of J_j and STD_j defines the sum of travel time of this job between the stations used to process the latter.

The objective of this choice is to give priority to jobs that have the value of long travel time over the processing time. Indeed, the purpose of this decision is to take advantage of the existence time of the jobs, with small values of Rtd_j, on the transport system in order to process the jobs which have the higher value of Rtd_j. The following table shows the data relating to the ratio Rtd_j, for the chosen example.

Table 4. Reports relating to processing times and job displacement

Jobs	$\boldsymbol{STD_j}$	$\boldsymbol{STT_j}$	$\boldsymbol{Rtd_j}$
$\boldsymbol{J_1}$	32	35	1.5
$\boldsymbol{J_2}$	16	53	3.3
$\boldsymbol{J_3}$	48	31	0.6
$\boldsymbol{J_4}$	32	16	0.5

From table 4, and according to the JSSPTtran heuristic, it is very clear that the job that must be processed first in the system is job J_4 which has the minimum value of Rtd_j. This job is followed by job $\boldsymbol{J_3}$ which therefore presents the second job to be processed in the system. Then job $\boldsymbol{J_1}$ and job $\boldsymbol{J_2}$. The solution selected for our problem is defined in this case by the order mentioned in the following table The C_{max} calculated from this sequencing is equal to 168.5 units of time.

Tab 5. Job order according to JSSPTtran

Order on machines	**Machines**			
	$\mathbf{M_1}$	$\mathbf{M_2}$	$\mathbf{M_3}$	$\mathbf{M_4}$
1	J_4	J_4	J_4	J_3
2	J_3	J_3	J_3	J_1

3	J_1	J_1	J_1	J_2
4	J_2	J_2	J_2	-

4.3 The heuristic JSLPTtran

Unlike the previous heuristic, the sequencing of jobs according to the JSLPTtran heuristic is done according to the decreasing order of Rtd_j. The objective here is to give priority to jobs that have higher processing time values than just displacement. Applying this heuristic to the illustrative example, allows us to obtain the jobs order mentioned in Table 6. This solution generates a value of C_{max} which is equal to 136 time units.

Tab 6. Job order according to JSLPTtran

Order on machines	Machines			
	$\mathbf{M_1}$	$\mathbf{M_2}$	$\mathbf{M_3}$	$\mathbf{M_4}$
1	J_2	J_2	J_2	J_2
2	J_1	J_1	J_1	J_1
3	J_3	J_3	J_3	J_3
4	J_4	J_4	J_4	-

4.4 The heuristic JSSRCtran

For this heuristic, we propose to introduce the number of consumable resources in the calculation of the choice ratio Rtd_j. The principle is to program the jobs according to the increasing order of ratio $RtdRC_j$ (equation 4) which is equal to Rtd_j multiplied by the quantity of resources to consume $Qdem_j^c$

$$RtdRC_j = Rtd_j * Qdem_j^c \quad (4)$$

This heuristic aims to place at the start of the schedule the jobs that consume the least components and that require lower processing times compared to travel times. Applying this heuristic to the same example gives the order shown in Table 7. The C_{max} obtained in this case is 168.5 units of time.

Tab 7. Job order according to JSSRCtran

Order on machines	Machines			
	$\mathbf{M_1}$	$\mathbf{M_2}$	$\mathbf{M_3}$	$\mathbf{M_4}$
1	J_4	J_4	J_4	J_3
2	J_3	J_3	J_3	J_1
3	J_1	J_1	J_1	J_2
4	J_2	J_2	J_2	-

4.5 The heuristic JSLRCtran

The choice of the order of the jobs according to this heuristic is based on the decreasing order of the ratio $RtdRC_j$, which gives priority to the jobs which consume more resources and which require more processing time than their movements. The following table shows the order of jobs after applying the JSLRCtran heuristic to our example. The C_{max} obtained here is 168 time units.

Tab 8. Job order according to JSLRCtran

Order on machines	Machines			
	$\mathbf{M_1}$	$\mathbf{M_2}$	$\mathbf{M_3}$	$\mathbf{M_4}$
1	J_2	J_2	J_2	J_2
2	J_1	J_1	J_1	J_1
3	J_3	J_3	J_3	J_3
4	J_4	J_4	J_4	-

4.6 The heuristic JSSPT cumtran

JSSPTcumtran is a heuristic that consists of scheduling, on each machine, the jobs in ascending order of their cumulative time ratios $Rtdcum_j^k$ (described by equation 5).

$$Rtdcum_j^k = \frac{STTcum_j^k}{STDcum_j^k} \quad (5)$$

This ratio is defined by dividing the cumulative processing time $STTcum_j^k$ over the cumulative travel time $STDcum_j^k$. The objective sought by this heuristic is to place in first the jobs which arrive and finish their executions as soon as possible.

From the data presented in Table 9 and according to equation 5 we can calculate the values of $Rtdcum_j^k$ which are shown in the following table

According to the heuristic JSSPTcumtran the solution of the example problem is presented in Table 11.

Tab 9. Valeurs Cumulées des temps de traitement et de déplacement des jobs

Jobs	M_1		M_2	
	$STTcum_j^1$	$STDcum_j^1$	$STTcum_j^2$	$STDcum_j^2$
J_1	5	17.5	10	22.5
J_2	15	1.5	25	6.5
J_3	26	33.5	21	22.5
J_4	13	17.5	9	6.5
Jobs	M_3		M_4	
	$STTcum_j^3$	$STDcum_j^3$	$STTcum_j^4$	$STDcum_j^4$
J_1	2	9.5	35	32
J_2	45	9.5	53	16
J_3	7	9.5	31	48
J_4	16	25.5	-	-

Table 10. The values of the ratio $Rtdcum_j^k$

Jobs	Machines			
	M_1	M_2	M_3	M_4
J_1	0.28	0.44	0.21	1.09
J_2	10	3.85	4.74	3.31
J_3	0.78	0.93	0.74	0.65
J_4	0.74	1.38	0.50	-

Tab 11. Job order according to JSSPTcumtran

Order on machines	Machines			
	M_1	M_2	M_3	M_4
1	J_1	J_1	J_1	J_3
2	J_4	J_3	J_4	J_1
3	J_3	J_4	J_3	J_2
4	J_2	J_2	J_2	-

4.7 The heuristic JSLPTcumtran

The decision of the choice of solution in the JSLPTcumtran heuristic is based on the decreasing order of ratio $Rtdcum_j^k$. According to this heuristic the sequencing of the jobs is carried out as follow :

Tab 11. Job order according to JSLPTcumtran

Order on machines	*Machines*			
	M_1	M_2	M_3	M_4
1	J_2	J_2	J_2	J_2
2	J_3	J_4	J_3	J_1
3	J_4	J_3	J_4	J_3
4	J_1	J_1	J_1	

5 RESULTS AND DISCUSSION

In order to study the performance of the proposed methods, we carried out several simulations of the problem on numerous instances. However, we have divided our analysis into two parts. The first part consists in evaluating the quality of the solutions obtained by using the proposed approaches compared to the optimal solution. The instances selected in this part are small instances. In the second part, the study was extended to medium and large instances. The objective is to find which approach is best suited to our problem and which gives the best solutions.

Indeed, according to our research, we could not find benchmarks used in the literature where the authors address exactly the same problem studied in this work. We, therefore, proposed to generate, in a random manner, benchmarks specific to this study (appendix A). Then, operating ranges, processing times, transport times as well as quantities of consumable resources are generated randomly using a program under Matlab. The random generation of this data was done for fifteen different examples from each instance.

5.1 Results obtained for small instances

In this part, we focused our study on three problems of small sizes 2x4, 3x4, and 4x4, of which we made a comparison between the heuristics proposed, and the exact method. The objective was to evaluate the performance of these heuristics and improve the quality of their solutions with respect to the lifetime of the optimal solution. Indeed, to obtain solutions to our problem, we have programmed all of the approaches in Matlab whose processing times are generated between 1 and 30 time units. The evaluation criterion used to compare the proposed approaches is the value of the objective function C_{max}. The results in terms of Makespan means (C_{max}) obtained by applying these approaches on 15 different examples are presented in figures 4, 5, and 6.

We notice from the results shown in these figures that the JSSPTcum heuristic gives results very close to the optimum for the 2x4 and 3x4 instances and the best solutions for the 4x4 instance, therefore it dominates all other heuristics. We can also notice that the heuristic JSSPTcumtran ranks second after JSSPTcum for all small-sized instances, thus surpassing the other five heuristics. For the five remaining heuristics, they have practically the same classification, with a small overrun of the heuristics JSLPTcum, JSSPTtran, and JSLPTtran, which do not include the number of consumable resources in the report of the choice, compared to the two heuristics JSSRCTtran and JSLRCTtran which are, in most of the time, in the last positions.

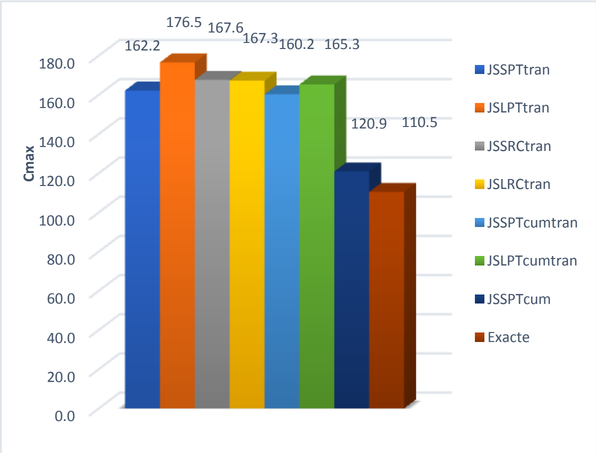

Fig. 4 Results obtained for 2*4 problem

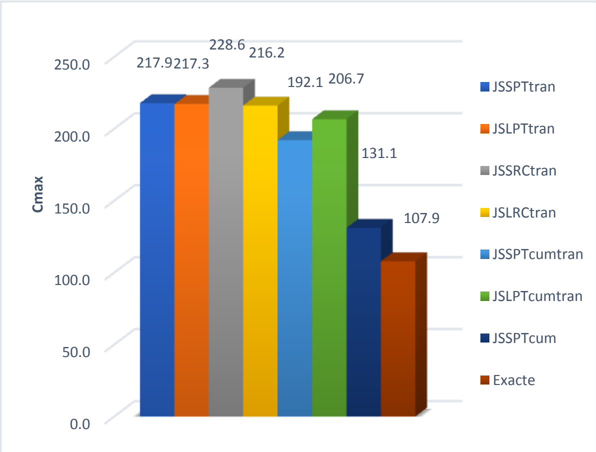

Fig. 5 Results obtained for 3*4 problem

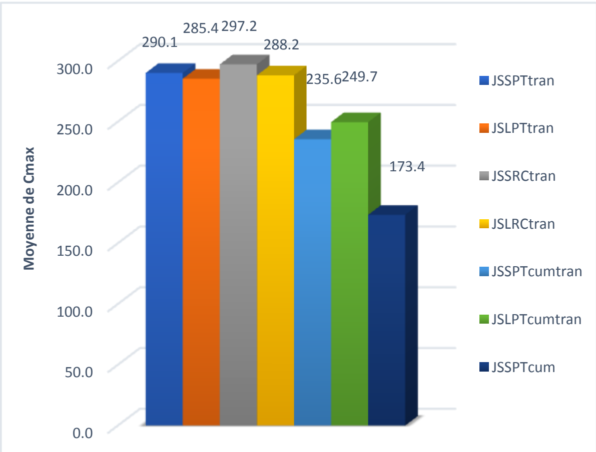

Fig. 6 Results obtained for 4*4 problem

In addition, we see that, between two heuristics using the same choice ratio, heuristics that are based on the increasing order of the ratio perform better than other heuristics that use decreasing order in the choice of solutions.

Through these results we can see, firstly, that the use of the cumulative value in the report of the choice of the solutions demonstrates a great efficiency concerning the identification of the sequencing of the jobs on the machines, this is well proved by the results of the two heuristics JSSPTcum and JSSPTcumtran. Second, the integration of quantities of consumable resources in the selection reports does not allow to find good solutions to the studied problem. Third, the increasing order of ratios presents a very efficient criterion for the choice of job sequencing and which often leads to better solutions for small instances of our problem.

5.2 Results obtained for medium and large instances

The objective of this part is to approve the efficiency of the heuristics proposed in solving the problem with different complexities. The choice of instances has been fixed as follows: 8x4, 10x4, and 15x4 for medium instances and 20x4, 50x4, and 100x4 for large instances. The study of the approaches in this part is also based on the evaluation of the Makespan values calculated through 15 different examples for each instance. The results obtained from all the tests carried out are presented in two figures 7 and 8.

These figures successively show the variation of C_{max} average calculated for the medium and large instances. Noting here that, in this part of the expertise, the number of admissible solutions found by the two heuristics JSSPTcumtran and JSLPTcumtran does not exceed two solutions out of the fifteen examples studied, which shows the weakness of these two heuristics concerning the problems of averages and large bodies. Therefore, we preferred to completely exclude their results from our comparison.

Through the results obtained after all the tests carried out, we notice very well that the use of the heuristic JSSPTcum gave very good results compared to the other heuristics which once again demonstrates its efficiency in solving the problem for larger instances. On the other hand, for the other heuristics and contrary to what is observed in the first part of the expert appraisal (the small instances), the heuristics which use the decreasing order of the ratio of choice of the solutions (JSLPTtran and JSLRCtran), exceed the others heuristics using the ascending order of the ratio (JSSPTtran and JSSRCtran), which proves and motivates their choices regarding the resolution of the problem for medium and large instances.

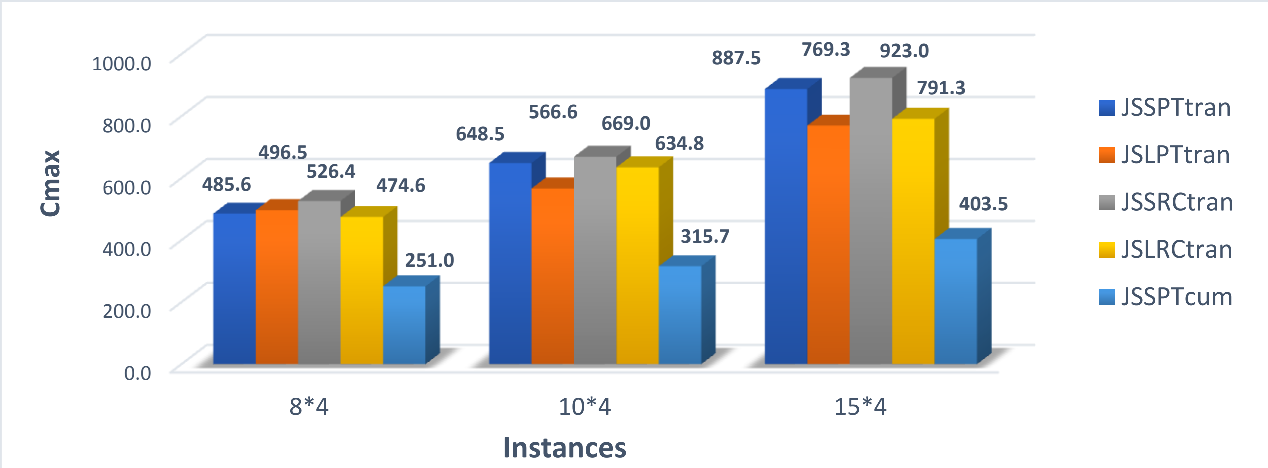

Fig. 7 Results obtained for medium instances

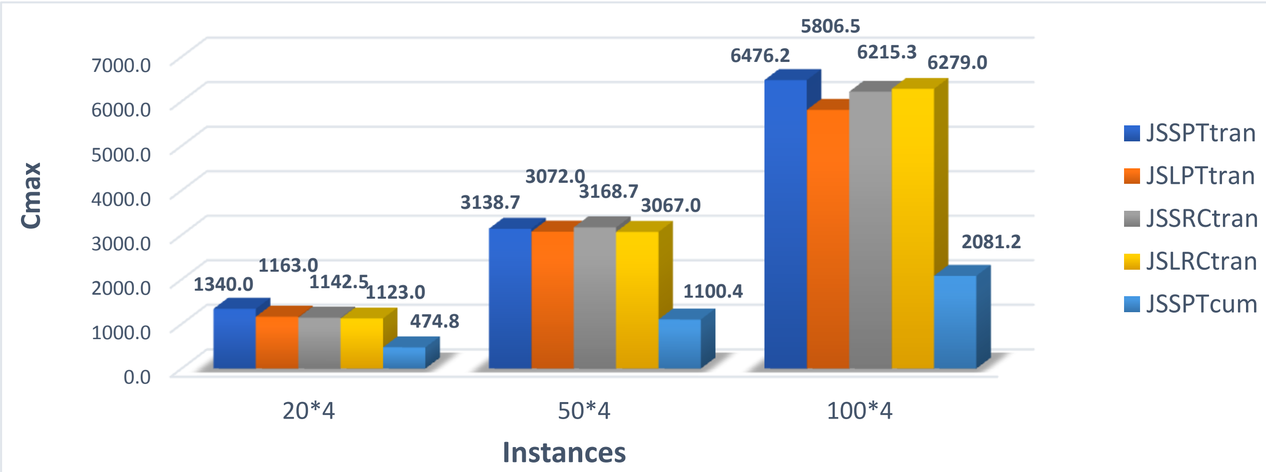

Fig. 8 Results obtained for large instances

6 CONCLUSION

In this paper, we are interested in solving the makespan minimization problem in a job shop environment with two constraints, unidirectional transport constraint, and non-renewable resource constraint. These constraints complicate the decision-making process for scheduling tasks, which may require multiple components of different types at the same time. We have shown the effectiveness of heuristics based on priority rules in solving this type of problem on a large number of different configurations. The comparison between these methods shows the advantage of the heuristic JSSPTcum over other approaches in terms of makespan.

The study carried out in this work allowed us to identify some preliminary results which give the possibility of evaluating the scheduling decision-making system using heuristic methods;

- The use of the cumulative value in the report of choice of solutions, presents a very effective tool concerning the identification of the sequencing of the jobs for the minimization of C_{max} of our problem.
- Concerning the heuristics which integrate the quantity of consumable resources in the report of choice are, in the majority of cases, less efficient than the other heuristics.
- We have found that the larger the problem size, the more efficient heuristics using the descending order of the ratio of choice become.

Finally, through the various studies that have been carried out, we have found that, the approach which remains the most efficient, it is the heuristic JSSPTcum which exceeds the other approaches concerning all the sizes of the problem

7 REFERENCES

Drăghici, Banharnsakun, A., Sirinaovakul, B., & Achalakul, T. (2012). *Job shop scheduling with the best-so-far ABC*. Engineering Applications of Artificial Intelligence, 25(3), 583-593.

Banharnsakun, A., Sirinaovakul, B., & Achalakul, T. (2012). *Job shop scheduling with the best-so-far ABC*. Engineering Applications of Artificial Intelligence, 25(3), 583-593.

Belkaid, F., Hadri, A., & Bennekrouf, M. (2018, April). *Efficient Approach for Parallel Machine Scheduling Problem. In 2018 International Colloquium on Logistics and Supply Chain Management* (LOGISTIQUA) (pp. 158-163). IEEE.

Caumond, A. (2006). *Le problème de job shop avec contraintes : modélisation et optimisation*. Doctoral dissertation, Université Blaise Pascal-Clermont-Ferrand II.

Dabah, A., Bendjoudi, A., AitZai, A., El-Baz, D., & Taboudjemat, N. N. (2018). *Hybrid multi-core CPU and GPU-based B&B approaches for the blocking job shop scheduling problem*. Journal of Parallel and Distributed Computing, 117, 73-86.

Delgado Sobrino, D. R., & Košťál, P. (2013). *A few analysis and customization issues of a new icim 3000 system: the case of the material flow, its complexity and a few issues to improve*. Academic Journal of Manufacturing Engineering, 11(4).

Deng, G., Su, Q., Zhang, Z., Liu, H., Zhang, S., & Jiang, T. (2020). *A population-based iterated greedy algorithm for no-wait job shop scheduling with total flow time criterion*. Engineering Applications of Artificial Intelligence, 88, 103369.

Đurasević, M., & Jakobović, D. (2018). *Comparison of ensemble learning methods for creating ensembles of dispatching rules for the unrelated machines environment*. Genetic Programming and Evolvable Machines, 19(1), 53-92.

Gondran, M., Huguet, M. J., Lacomme, P., Quilliot, A., & Tchernev, N. (2018). *A Dial-a-Ride evaluation for solving the job-shop with routing considerations.* Engineering Applications of Artificial Intelligence, 74, 70-89.

Gong, G., Deng, Q., Chiong, R., Gong, X., & Huang, H. (2019). *An effective memetic algorithm for multi-objective job-shop scheduling*. Knowledge-Based Systems, 182, 104840.

Grabowski, J., & Janiak, A. (1987). *Job-shop scheduling with resource-time models of operations*. European Journal of Operational Research, 28(1), 58-73.

Heger, J., & Voß, T. (2019). *Dynamic priority based dispatching of AGVs in flexible job shops*. Procedia CIRP, 79, 445-449.

Homayouni, S. M., & Fontes, D. B. (2019, February). *Joint scheduling of production and transport with alternative job routing in flexible manufacturing systems*. In AIP Conference Proceedings (Vol. 2070, No. 1, p. 020045). AIP Publishing LLC.

Kassu, J., & Eshetie, B. (2015). *Job Shop Scheduling Problem for Machine Shop with Shifting Heuristic Bottleneck*. Global Journal of Research In Engineering.

Li, Y., Shi, S. Y., & Huang, Q. D. (2018). *Three-machine job shop scheduling with intermediate transfer*. International Journal of Simulation Modelling (IJSIMM), 17(2), 359-368.

Lin, T. L., Horng, S. J., Kao, T. W., Chen, Y. H., Run, R. S., Chen, R. J., ... & Kuo, I. H. (2010). *An efficient job-shop scheduling algorithm based on particle swarm optimization*. Expert Systems with Applications, 37(3), 2629-2636.

Ozturk, G., Bahadir, O., & Teymourifar, A. (2019). *Extracting priority rules for dynamic multi-objective flexible job shop scheduling problems using gene expression programming*. International Journal of Production Research, 57(10), 3121-3137.

Rajendran, S., Rajendran, C., & Leisten, R. (2017). *Heuristic rules for tie-breaking in the implementation of the NEH heuristic for permutation flow-shop scheduling*. International Journal of Operational Research, 28(1), 87-97.

Tabrizi, B. H., Ghaderi, S. F., & Haji-Yakhchali, S. (2019). *Net present value maximisation of integrated project scheduling and material procurement planning*. International Journal of Operational Research, 34(2), 285-300.

Toker, A., Kondakci, S., & Erkip, N. (1994). *Job shop scheduling under a non-renewable resource constraint*. Journal of the Operational Research Society, 45(8), 942-947.

Zhang, J., Ding, G., Zou, Y., Qin, S., & Fu, J. (2019). *Review of job shop scheduling research and its new perspectives under Industry 4.0.* Journal of Intelligent Manufacturing, 30(4), 1809-1830.

Zhou, D.N., Cherkassky, V., Baldwin, T.R. and Olson, D.E. (1991). *A neural network approach to job-shop scheduling.* IEEE Transactions on Neural Networks, 2(1), 175–179.

Zhou, Y. D., Shin, J. H., & Lee, D. H. (2019). *Loading and scheduling for flexible manufacturing systems with controllable processing times*. Engineering Optimization, 51(3), 412-426.