Classification Rules for the Job Shop Scheduling Problem with Machine Breakdowns

N. Said, W. Mouelhi, and K. Ghedira

Abstract—**This paper addresses the job shop scheduling problem in the presence of machine breakdowns. In this work, we propose to exploit the advantages of data mining techniques to resolve the problem.**

We proposed an approach to discover a set of classification rules by using historic scheduling data. Intelligent decisions are then made in real time based on this constructed rules to assign the corresponding dispatching rule in a dynamic job shop scheduling environment.

A simulation study is conducted at last with the constructed rules and four other dispatching rules from literature. The experimental results verify the performance of classification rule for minimizing mean tardiness.

Index Terms—**Job shop scheduling, classification rules, simulation, machine breakdowns.**

I. Introduction

Scheduling is defined as the allocation of resources to jobs over time. It is a decision-making process with the goal of optimizing one or more objectives.

Most of job shop scheduling approaches usually focus on a static environment and assume deterministic and known in advance data. However, in real-world applications, this situation is not often met since data may be subject to uncertainty and it may change over time. Such an approach is called offline scheduling.

Approaches that consider unexpected events that can change the system status and affect performance are called online scheduling. Examples of such online events include machine breakdowns, arrival of urgent jobs, due date changes, etc.

Online scheduling has been solved using different techniques such as metaheuristics, heuristics, artificial intelligence techniques, multi-agent systems and dispatching rules.

Machine breakdowns are considered as one of the most challenging issues in production scheduling. The easiest solution, in case of an unexpected breakdown, is to use some dispatching rules which are simple heuristics aimed at selecting the next job immediately after the breakdown occurs [1].

This paper contributes to the literature by developing a set of classification rules using historical scheduling data to cope with the job shop scheduling problem with machine breakdowns. The constructed classification rules are then compared with traditional dispatching rules from literature via simulation experiments.

The rest of this paper is organized as follows. A literature review of related research is presented in Section II. In Section III we provide a description of the proposed approach. Section IV provides details of the studied job shop configuration and simulation model used in the experiments. In Section V, the detailed experimental results are presented, followed by the conclusions and future research suggestions in Section VI.

II. Literature Review

Numerous works have been performed to solve the job shop scheduling problem in the presence of machine breakdown using different techniques: metaheuristics, artificial intelligence techniques [2]-[4], multi-agent systems [5]-[7], and dispatching rules.

In this paper we focus essentially on dispatching rules developed and used for the dynamic job shop problem. We focus also on data mining based approach developed to deal with the job shop scheduling problem in the presence of machine breakdowns.

In the next section, we give an overview of the different approaches proposed in literature to deal with this problem.

A. Dispatching Rules Developed for the Dynamic Job Shop Problem

A limited number of published works using dispatching rules in the presence of machine breakdowns exist in literature.

An experimental comparison between different combinations of dispatching rules is provided in [8] for the case of dynamic job shop with machine breakdowns. It evaluated the effects of several breakdown levels on performance of dispatching rules. Their results showed that, with respect to the due date-based objectives, dispatching rules were more affected by the breakdown level and the mean time to repair.

In [9], the authors tested the performance of several schedule repair heuristics based on rerouting the jobs to their alternative machines when their primary machine fails.

In [10], a comparative study has been made of tardiness based existing dispatching rules as well as new dispatching rules. Mean tardiness, maximum tardiness and the number of tardy jobs objectives have been used to evaluate the performance of each dispatching rule. Several experimental parameters related to shop loading levels, breakdown levels and mean time to repair, are taken into consideration to

Manuscript received September 9, 2014; revised November 24, 2014.

N. Said and K. Ghedira are with the Higher Institute of Management of Tunis (University of Tunis), Tunisia (e-mail: said.nesrine@gmail.com, khaled.ghedira@isg.rnu.tn).

W. Mouelhi is with the Higher Institute of Computer Science (University of Tunis ElManar), Tunisia (e-mail: wiem_mouelhi@yahoo.fr).

DOI: 10.7763/IJIEE.2015.V5.548

analyze the effect of these parameters on the performance of the various dispatching rules. Simulation results indicate that shop loading, breakdown level and mean time to repair are important parameters for the selection of an appropriate rule in a shop.

In [11], the authors provided a comparative study of dispatching rules for the case of dynamic flexible job shop scheduling problem with sequence-dependent setup times and machine breakdowns. The used rules comprise three new routing rules taking into account the machine breakdown, and two routing rules from the literature of dynamic flexible job shop scheduling problem.

B. Data Mining Based Approach to Job Shop Scheduling

The review of the literature reveals that machine learning has used for the first time in production scheduling to select the best dispatching rule using simulated data.

Learning can be used on production scheduling to discover a new dispatching rule and also to improve an existing one. Explicit and implicit knowledge can be extracted from a set of good schedules. Many correlations between system attributes can be discovered from historical data and can be used as training data for a learning algorithm.

To the best of our knowledge, almost the published works using data mining for solving the job shop scheduling problem consider a static environment without the presence of machine breakdowns.

In [12], the authors used an induction algorithm to generate a set of rules for a flow shop problem. In [13], the authors used also a learning algorithm which has for goal to generate a set of attributes for on-line rule selection.

In [14], authors applied data mining techniques on data generated by a genetic algorithm based scheduling and developed a rule set approximating the genetic algorithm scheduler. In [15] and [16], authors used genetic programming to generate sequencing policies in the form of dispatching rules.

A data mining based approach is provided in [17] to generate dispatching rules by applying a decision tree algorithm to historical data to discover the key scheduling concepts.

In [18], the authors developed a new approach to generate the knowledge of systems behavior in a simulated job shop based on the combination of an evolutionist optimization and an induction graph learning approach.

A novel approach to discover new dispatching rules based on data mining techniques was proposed in [19]. A new approach for discovering previously unknown priority dispatching rules using data mining technique was also proposed in [20] and [21].

A new approach was developed in [22] to find out dispatching sequence in the form of IF-Then-Else rules for two machine flow shop scheduling using decision tree approach.

In [23] the authors proposed a solution to discover scheduling rules for a job shop problem using Decision Tree algorithm.

III. Proposed Approach

As we have mentioned above, we propose a data mining based approach to cope with the dynamic job shop problem using historical scheduling data.

The first goal of our work is to construct a set of classification rules. Classification rules have the advantage of representing knowledge at a high level of abstraction; therefore they are intuitively comprehensible to the user [24].

Classification rule learners work on training data and aims to construct the smallest rule set consistent with the historic scheduling data. More precisely, given a set of training examples, classification rule learning task aims at finding a set of classification rules that can be used to predict or classify new cases that haven't been presented to the learner before.

Therefore, the first goal aims at inducing a model by using a set of previously known examples called training instances. Thus, we should start by gathering a large collection of examples of job shop states.

The learning algorithm assumes that the training instance has a fixed set of predicting attributes and a single objective containing the class label of each training example.

For instance in a job shop problem, an example might be described by attributes such as the mean time to repair, the breakdown level, the average due date factor, etc.

The labels are the possible dispatching rules that we are trying to predict.

The set of training example (candidate rules) is implemented in a discrete event simulation model of the job shop.

The performance of the rules is compared using simulation experiments under varying values of system attributes characterizing the job shop dynamics, and the rules which are taken as training examples (input to the learning system) are those having the best performance under variety of system states.

The process starts then with the choice of the specific learning algorithm to be used. This algorithm is then applied to all examples in the training set. It analyzes the relationship between the predictor attributes and the class attributes for all training examples, and discovers a model that best describes the dataset.

Classification rules constructed from training data are represented in the IF-THEN form described as:

IF < conditions> THEN < class>

In each rule, the <condition> part contains a logic combination of predictor attributes in the form: $term_1$ and $term_2$ and $term_i$…

Each term is a triple < attribute, operator, value>. The consequent part of conclusion <class> shows the predicted class whose cases satisfy the <condition> part of the rule.

Constructed rules are then used to take intelligent decision in real time to assign the corresponding dispatching rule given a system state. Classification rules should then be able to generalize from the presented data to unseen examples.

In this research, the objective considered is the minimization of mean tardiness.

IV. Experimental Settings

We consider the following job shop [25] consisted of the use of four machines, M1, M2, M3, and M4 to manufacture four types of orders. Jobs arrive continuously according to an exponential distribution with a mean of 5.5 time units (TU). The processing time of operations on the various machines and machine type (mean, standard deviation, in TU) is presented in Table I.

The processing sequence of each order is presented in Table II.

We used Arena simulation software version 14 to develop the simulation model for the job shop scheduling problem used in our work.

To construct the classification rule set, we used discrete event simulation.

We started first by constructing the benchmark then applying the C4.5 algorithm [26] to generate classification rules.

The dispatching rules used to construct the benchmark are chosen from literature (FIFO, SPT, EDD, and MST).

FIFO (First In, First Out) Using the FIFO rule, the jobs are processed in the order they arrive at the machine.

SPT (Shortest Processing Time) Using the SPT rule, the job with the shortest processing time is selected.

EDD (The Earliest Due Date) Using the EDD rule, the next job to be processed is the one with the earliest due date.

MST (Minimum Slack Time rule) Using the MST rule, the job with the minimum slack time is scheduled. Slack time of a job is computed by deducting the current time and the total remaining process time from the due date of the job.

The literature review reveals that dispatching rules performance in the presence of machine breakdown mostly depends on several factors such as changes in arrival rates, duration and frequency of the machine failures, and average due date factor.

For this reason, in our work we choose to describe a state using a set of predictor attributes which constitute the <condition> part of the classification rule.

The chosen attributes are: the Mean Time To Repair (MTTR), the Mean Time Between Failure (MTBF), the average Due date factor (tightness factor), and the arrival rates.

The label is the dispatching rule that we are trying to predict.

The rules which are taken as training examples (input to the learning system) are those having the minimal mean tardiness under variety of system states.

TABLE I: PROCESSING TIMES PER JOB AND MACHINE TYPE (MEAN, STANDARD DEVIATION, IN TU)

Job type	Proportion in arrivals (%)	M 1	M 2	M 3	M 4
1	40	[9,1.3]	[7,1]	[8,1.2]	_
2	15	[7,1]	[8,1.2]	[9,1.3]	[7,1]
3	20	_	[9,1.3]	[7,1]	[9,1.5]
4	25	[9,1.5]	[4,1]	_	[8,1]

TABLE II: PROCESSING SEQUENCE

Job type	Sequence
1	M1 – M3 – M2
2	M2- M3 - M1 – M4
3	M2- M3 - M4
4	M4- M2- M1

V. COMPUTATIONAL RESULTS

In order to analyze the relative performance of the constructed rules with respect to the minimization of mean tardiness, we used the same job shop [25] used for the construction of the benchmark.

We consider different scenarios of machine breakdowns under varying values of MTTR, MTBF, due date factor, and also changing the unavailable machine.

The results of the simulation study are obtained by taking the mean of mean values of 7 replications.

The MTTR and MTBF are exponentially distributed.

Fig. 1 to Fig. 7 show the mean tardiness for all rules with respect to different breakdown levels and Mean time to repair.

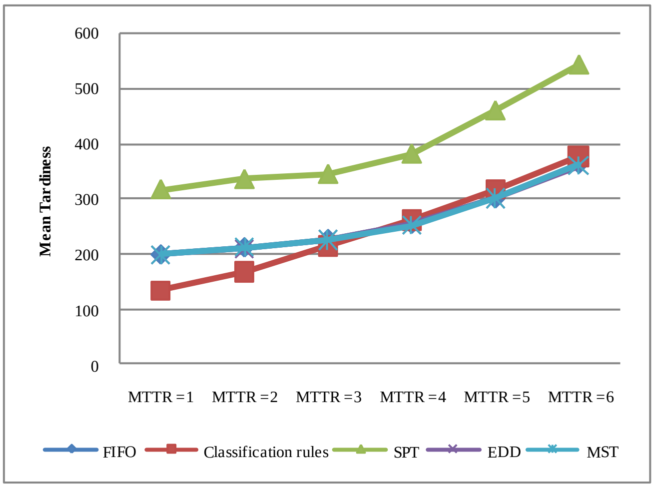

Fig. 1. Classification rules performance (M1 failed, tight due date, MTBF= expo (10)).

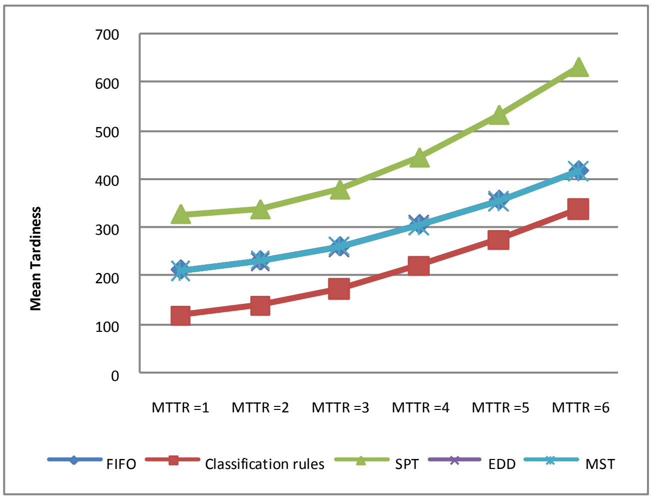

Fig. 2. Classification rules performance (M1 failed, tight due date, MTBF= expo(20)).

For most of the times, classification rules are significantly better or comparable to the best performing rule for the mean tardiness measure.

The obtained results proved also that classification rules performance in the presence of machine breakdown depends on MTTR level, MTBF level, and average due date factor.

Classification rules are significantly better than the performance of all the other rules in minimizing mean tardiness with tight and loose due date.

For tight and loose due date, the classification rules performance is significantly better than all other rules for minimizing mean tardiness with low level of MTBF and low MTTR level.

With high level of MTTR, FIFO, EDD, and MST rules outperform classification rules, and the differences between them are minor (Fig. 1 and Fig. 3).

With high level of MTBF, classification rules are significantly better than all other rules for minimizing mean tardiness.

The results revealed also that the relative performance of classification rules can be affected by changing the broken machine (Fig. 4 to Fig. 7). Classification rules are significantly better than the performance of all the other rules in minimizing mean tardiness with tight and loose due date for the different other broken machines in the used job shop.

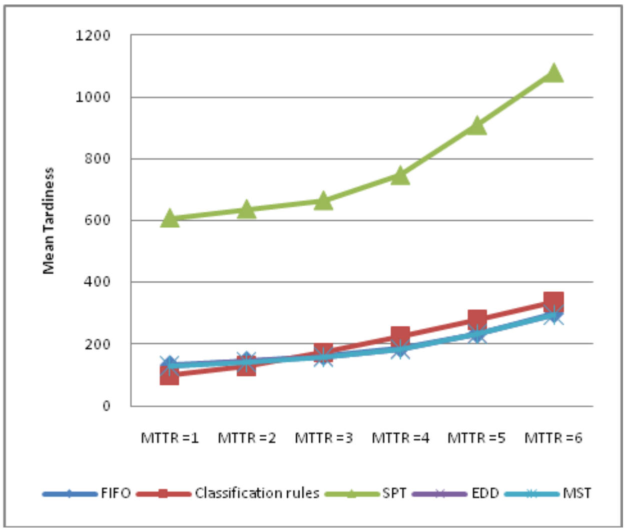

Fig. 3. Classification rules performance (M1 failed, loose due date, MTBF=expo(10)).

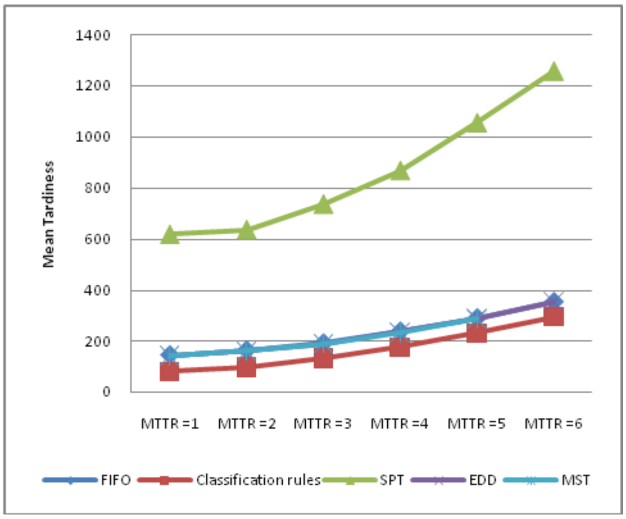

Fig. 4. Classification rules performance (M1 failed, loose due date, MTBF= expo(20)).

VI. Conclusions and Future Works

In this paper, we discuss the use of classification rules to solve the job shop scheduling problems considering unexpected machine breakdowns.

The goal of our work aims at inducing a set of classification rules by using a set of historical scheduling data. Constructed rules are then used to take intelligent decision in real time to assign the corresponding dispatching rule given a new state of the system.

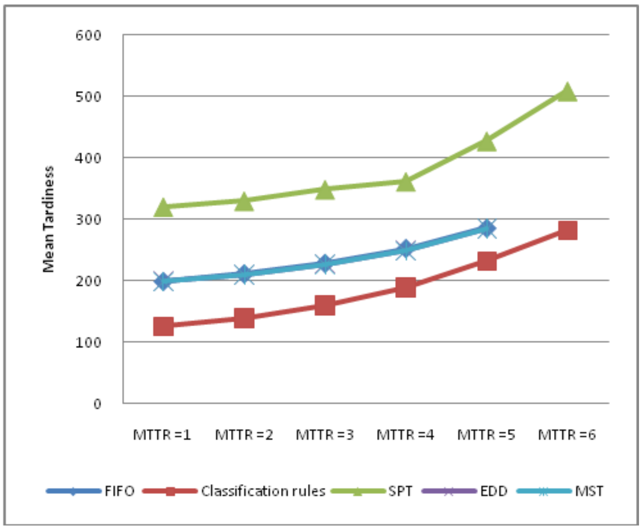

Fig. 5. Classification rules performance (M2 failed,tight due date, MTBF= expo(10)).

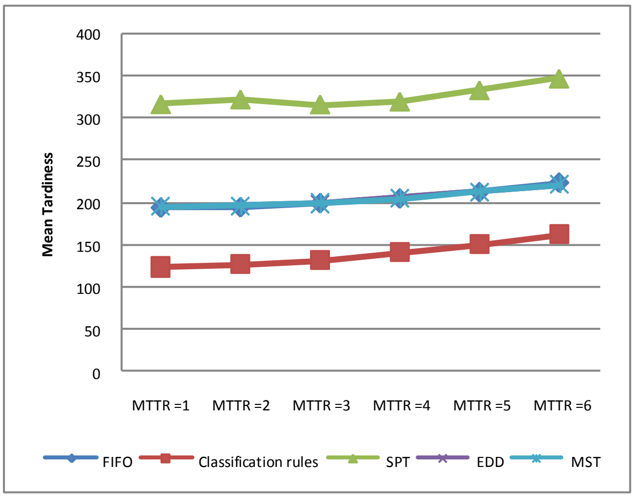

Fig. 6. Classification rules performance (M3 failed, tight due date, MTBF= expo(10)).

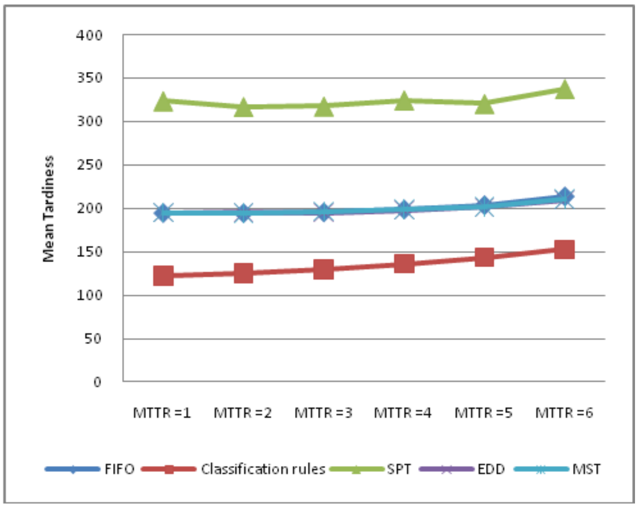

Fig. 7. Classification rules performance (M4 failed, tight due date, MTBF= expo(10)).

The performance of the rules is compared using simulation experiments under varying values of system attributes characterizing the job shop dynamics.

It is observed that the results of classification rules are always superior or at least comparable to the best performing rule for the mean tardiness measure.

We believe that it is possible to more improve the performance of classification rule set, which may be obtained through a better selection of predictor attributes.

It is also good to extend the constructed benchmark with new instances of the job shop scheduling problem.

REFERENCES

[1] J. H. Blackstone, D. H. Phillips, and G. L. Hogg, "A state of the art survey of dispatching rules for manufacturing job shop operations," *International Journal of Production Research*, vol. 20, pp. 27-45, 1982.

[2] H. Suwa and H. Sandoh, "Capability of cumulative delay based reactive scheduling for job shops with machine breakdowns," *Computers and Industrial Engineering* , vol. 53, pp. 63-78, 2007.

[3] R. Zhou, A. Y. C. Nee, and H. P. Lee, "Performance of an ant colony optimization algorithm in dynamic job shop scheduling problems," *International Journal of Production Research*, vol. 47, pp. 2903-2920, 2009.

[4] S. M. K. Hasan, R. Sarker, and D. Essam, "Genetic algorithm for jobshop scheduling with machine unavailability and breakdowns," *International Journal of Production Research*, vol. 49, pp. 4999-5015, 2011.

[5] K. Kang, Y. Q. Yang, R. F. Zhang, and Y. Q. Yang, "MAS equipped with ant colony applied into dynamic job shop scheduling," *Advanced Intelligent Computing Theories and Applications with Aspects of Artificial Intelligence*, vol. 4682, pp. 823-835, 2007.

[6] W. Xiang and H. P. Lee, "Ant colony intelligence in multi-agent dynamic manufacturing scheduling," *Engineering Applications of Artificial Intelligence*, vol. 21, pp.73-85, 2008.

[7] A. Rajabinasab, and S. Mansour, "Dynamic flexible job shop scheduling with alternative process plans: an agent-based approach," *The International Journal of Advanced Manufacturing Technology*, vol. 54, pp. 1091-1107, 2011.

[8] O. Holthaus, "Scheduling in job shops with machine breakdowns: An experimental study," *Computers and Industrial Engineering*, vol. 36, pp.137- 162, 1999.

[9] E. Kutanoglu and I. Sabuncuoglu, "Routing-based reactive scheduling policies for machine failures in dynamic job shops," *International Journal of Production Research*, vol. 39, pp. 3141-3158, 2001.

[10] A. Singh, N. K. Mehta, and P. K. Jain, "Tardiness based new dispatching rules for shop scheduling with unreliable machines," *International Journal of Simulation Modeling*, vol. 4, pp. 5-16, 2005.

[11] M. A. Shalaby, T. F. Abdelmaguid, and Z. Y. Abdelrasol, "New routing rules for dynamic flexible job shop scheduling with sequence-dependent setup times," in *Proc. the International Conference on Industrial Engineering and Operations Management*, pp. 747-756, Turkey , 2012.

[12] H. Pierreval and H. Ralambondrainy, "Generation of knowledge about the control of a flow shop using data analysis oriented learning techniques and simulation," Technical Report, p. 897, 1988.

[13] S. Nakasuka and T. Yoshida, "Dynamic scheduling system utilizing machine learning as a knowledge acquisition tool," *International Journal of Production Research*, pp. 411-431, 1992.

[14] D. A. Koonce and S. C. Tsai, "Using data mining to find patterns in genetic algorithm solutions to a job shop schedule," *Computers and Industrial Engineering*, vol. 38, pp. 361-374, 2000.

[15] C. Dimopoulos and A. M. S. Zalzala, "Investigating the use of genetic programming for a classic one-machine scheduling problem," *Advances in Engineering Software*, vol. 32, pp. 489-498, 2001.

[16] H. N. Binh and T. J. Cing, "Evolving dispatching rules for solving the flexible job-shop problem," in *Proc. IEEE Congress on Evolutionary Computation*, vol. 3, pp. 2848-2855, 2005.

[17] X. Li and S. Olafsson, "Discovering dispatching rules using data mining," *Journal of Scheduling*, vol. 8, pp. 515-527, 2005.

[18] A. L. Huyet, "Optimization and analysis aid via data-mining for simulated production systems," *European Journal of Operational Research*, vol. 173, pp. 827-838, 2006.

[19] C. D. Geiger, R. Uzsoy and H. Aytug, "Rapid modeling and discoveriny of priority dispatching rules: An autnomous learning approach," *Journal of Scheduling*, vol. 9, pp. 7-34, 2006.

[20] A. Shahzad and N. Mebarki, "Discovering dispatching rules for job shop scheduling problem through data mining," in *Proc. the International Conference of Modeling and Simulation* ,Tunisia, 2010.

[21] A. Shahzad and N. Mebarki, "Data mining based job dispatching using hybrid simulation-optimization approach for shop scheduling problem," *Engineering Applications of Artificial Intelligence*, vol. 25, pp.1173-1181, 2012.

[22] R. Balasundaram, N. Baskar, and R. S. Sankar, "A new approach to generate dispatching rules for two machine flow shop scheduling using data mining," in *Proc. the International Conference on Modeling Optimization and Computing*, vol. 38, pp. 238-245, 2012.

[23] R. Balasundaram, N. Baskar, R. Siva Sankar, "Discovering dispathcing rules for job shop schdeuling using data mining," *Advances in Intelligent Systems and Computing* , vol. 178, pp. 63-72, 2013.

[24] I. H. Witten and E. Frank, "Data Mining: Practical machine learning tools and techniques," Morgan Kaufmann, 2005.

[25] D. J. Fonseca and D. Navaresse, "Artificial neural networks for job shop simulation," *Advanced Engineering Informatics*, vol. 16, pp. 241-246, 2002.

[26] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufman, 1993.

N. said is a PhD student in computer science at the Higher Institute of Management (Tunisia). She received her master's degree in computer science in 2010 from the Higher Institute of Management (Tunisia). She is a member of the Research Laboratory SOIE (Tunisia). Her research interests include machine learning, artificial intelligence and production systems.

W. Moulehi received her master's degree in computer science in 2005 from INSA (France) and her PhD degree in 2009 from Blaise Pascal University (France). She is a member of the Research Laboratory SOIE (Tunisia). Her research interests include artificial Intelligence and production systems.

K. Ghedira has received the specialized engineer degree in computer science and applied mathematics from ENSIMAG (France), the engineer degree in hydraulic from ENSEEIHT (France), both the master and the PhD degrees in artificial intelligence from ENSAE (France), and finally the habilitation degree in computer science from the habilitation degree in computer science from ENSI in 1999 (Tunisia). He was a fellow research at the Institute of Computer Science and Artificial Intelligence of Neuch tel (Switzerland 1992-96) and an expert consultant at British Telecom (England 1995) and from April 2011, the general managing director of the Tunis Science City.

He is a professor at ISG (University of Tunis), the founding president of the Tunisian Association of Artificial Intelligence (ATIA-Tunisia), the founder and ex-director of the research laboratory SOIE (Tunisia).