An appropriate pattern to solving a parallel machine scheduling by a combination of meta-heuristic and data mining

Mohamadreza Kaviani [1,2], Majid Aminnayeri [1✉], Seyed Nima Rafienejad [1], Fariborz Jolai [3]

[1] Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran
kaviani.mr@aut.ac.ir, ✉ mjnayeri@aut.ac.ir, sn.rafienezhad@aut.ac.ir
[2] Department of Industrial Engineering, Islamic Azad University, North Tehran branch, Tehran, Iran
[3] Department of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran
fjolai@ut.ac.ir

Abstract: Scheduling problems' applications in nowadays competitive world and the rate of their result's usage in industrial area demonstrate the importance of this problem. In this article, an identical parallel machine by the objective function of total weighted tardiness is considered. As the *NP*-hard nature of these problems, meta-heuristic algorithms are commonly applied to solve them. These kinds of algorithms are able to reach the near optimal solution in an acceptable time, but do not explain how a solution developed. It is tried, by Attribute-Oriented Induction and Clustering technique, to reveal proper patterns relied under these problems characteristics and with that justify the final solution found by Particle Swarm Optimization (PSO) algorithm. Moreover, these rules could applied to similar problems and provide solutions that are generally better than simple dispatching rules.

[Mohamadreza Kaviani, Majid Aminnayeri, Seyed Nima Rafienejad, Fariborz Jolai. **An appropriate pattern to solving a parallel machine scheduling by a combination of meta-heuristic and data mining.** Journal of American Science 2012; 8(1):160-167]. (ISSN: 1545-1003). http://www.americanscience.org.

Keywords: Parallel Machine Scheduling, Particle Swarm Optimization, Data Mining, Clustering, Validity Index

1. Introduction

A parallel machine scheduling problem can be classified into three different types, identical machines, different speeds or uniform machines and completely unrelated machines. Also several performance measures such as Total Completion Time (*makespan*), Total Flow Time and Total Tardiness can be considered for the mentioned problems. In this article, an identical parallel machine with total weighted tardiness minimization objective function is under study.

Parallel machine problem could be introduced as n nonpreemptive jobs are simultaneously available at time zero that have to be processed on at most one of m machines. Each job i has an integer processing time p_i, a distinct due date d_i, a q_i penalty cost for tardy jobs and also job i immediately run after job j. For a given processing order of the jobs, the earliest completion time C_i and the tardiness T_i can be computed for each job, where tardiness is defined as $T_i = max\{0, C_i - d_i\}$. The objective is to find the processing order of the jobs that minimizes the sum of the weighted tardiness of all the jobs (Baker 1974).

According to complexity theory, meta-heuristics are common algorithms to solve *NP*-hard problems such as scheduling problems (Poursalik and Miri-Nargesi (2011), Pratchayaborirak and Kachitvichyanukul (2011) and Udomsakdigool and Khachitvichyanukul (2011)), and whereas parallel machine with total tardiness objective function belongs to *NP*-hard problems (Sun and Wang, 2003), meta-heuristics should be applied to solve them. But researchers do not gain any scene about the process of problem solving and the final solution (Koonce and Tsai, 2000). In order to relieve this problem and black box a combination of meta-heuristics and data mining algorithm is applied in this article.

2. Background

Koonce and Tsai (2000) proposed an algorithm to solve a job shop problem which used attribute oriented induction which is one of the data mining techniques. After choosing four different attribute for a job shop problem, authors investigate the efficiency of attributes in rule extraction for a sample problem with six jobs and six machines. They show the proposed algorithm however could not defeated the Genetic Algorithm (GA) but easily could perform better than other common procedures such as Shortest Processing Time (SPT). Weckman et al. (2008) continued the former article by using the Neural Network (NN) instead of attribute oriented induction.

Kumar and Rao (2009) changed the problem area to the batch processing flow line and also applied Ant Colony Optimization (ACO) instead of GA. they used the same attributes exactly as those used previously by Weckman et al. (2008).

In 2005, Sha and Liu (2005) assigned the due date in a job shop by the aid of decision tree technique. Ozturk et al. (2006) implemented the regression tree to estimate order time in a flow shop problem in 2006 and effect examination of data mining to design the production system by attribute oriented induction is conducted by Kusiak and Smith (2007). Casillas et al. (2009) and Yang and Wang (2008) approved that data mining techniques are able to improve the convergence rate of GA. Olafsson and Li (2010) developed a two stage algorithm for rule extraction from previous sequences by means of decision tree. Khademi Zare and Fakhrzad (2011) present a new algorithm to solve a flexible flow shop scheduling problem with fuzzy due dates by objective of minimizing total tardy jobs. They promote the GA by extracting clandestine relations between previous iteration solutions to reach near optimal sequences.

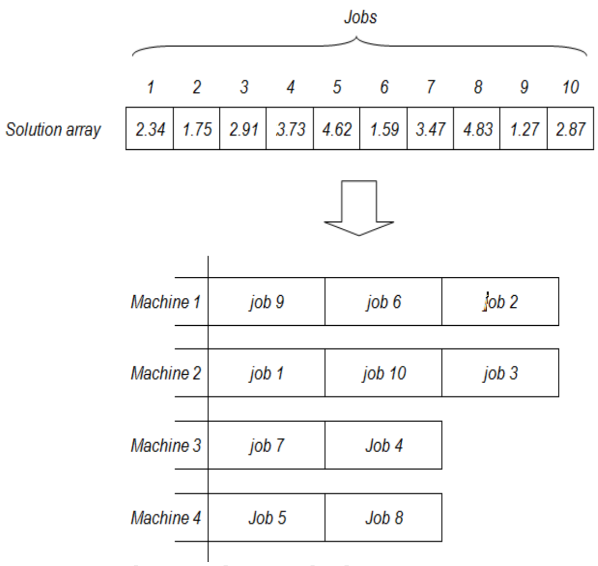

Figure 1: The encoding method

3. Proposed algorithm

The proposed algorithm contains five steps. The main part of the procedure fed on the answers of the PSO algorithm to extract appropriate patterns from those solutions. These steps are presented subsequently.

3.1. PSO

According to literature, GA is the most common algorithm to solve scheduling problems, but Niu et al. (2010) after study 250 sample problems in their article indicate that Particle Swarm Optimization (PSO) results better solution than GA for parallel scheduling problem with identical machine.

In this paper, a standard PSO with star neighborhood structure is applied to a parallel machine scheduling problem with 20 jobs and 4 machines shown in table 1 (generated by the standard method proposed by Fisher (1976)) to minimize total weighted tardiness.

Table 1. The instance problem

Job	Processing Time	Due Date	Penalty
1	5	14	1
2	9	11	2
3	16	30	1
4	7	17	4
5	23	31	6
6	11	28	1
7	1	5	7
8	6	27	3
9	8	16	5
10	15	25	3
11	19	26	6
12	2	15	4
13	4	8	8
14	4	14	6
15	9	21	10
16	9	14	4
17	2	12	7
18	14	22	2
19	15	26	1
20	5	9	1

There are different types of encoding methods, Kashan and Karimi (2009), Serifoglu and Ulusoy (1999) and Lee et al. (2006) are samples of solution representation. A new encoding method based on real numbers presented by Niu et al. (2010) is used in this paper. For mentioned problem with n job and m machine, n real numbers in the interval of *[1, m+1)* illustrate a solution in a solution array with n cells. The integer part of a real number denotes the machine that the job is assigned to and the fractional part represents the processing order of jobs on each machine. Based on the real number encoding, the sequence of jobs on each machine can be extracted from ascending order of fractional parts. Uniform distribution is used to produce n real number in the mentioned interval for the first generation randomly.

In the PSO algorithm an initial population (*P*) made-up of *N* particle are required that each individual particle defined by a specific position ($\vec{x_i}(t)$) and velocity ($\vec{v_i}(t)$). The initial velocity function contains a ($n \times$ *generation size*) matrix with the value of *zero* in all cells ($\vec{v_0} = 0$). These functions are updated subsequently according to problem information via Eq. (1) to Eq. (4) based on index *t*, which refers to the iteration number of the algorithm.

$$\vec{x_i}(t) = \vec{x_i}(t-1) + \vec{v_i}(t) \quad (1)$$

$$\vec{v}_i(t) = \varphi * \vec{v}_i(t-1) + \rho_1 \left(\vec{x}_{pbest_i} - \vec{x}_i(t-1)\right) + \rho_2 \left(\vec{x}_{pbest} - \vec{x}_i(t-1)\right) \quad (2)$$

$$\rho_1 = r_1 c_1 \quad (3)$$

$$\rho_2 = r_2 c_2 \quad (4)$$

In the mentioned equations, x_{pbest_i} represents the best obtained position for ith particle so far. Also, x_{gbest} denotes the best position among all the particles. The value of the first and second acceleration constants (c_1 and c_2) and the inertia weight (φ) are tuned to be 2 and 0.2 respectively. Moreover, the interval of *[1, m+1)* is employed as boundary constraint to particle position to guarantee that the variables are in the feasible region.

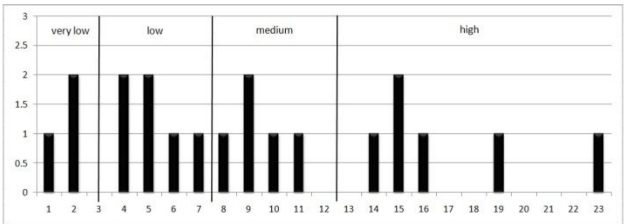

Figure 2. Clustering on frequencies of processing time attribute

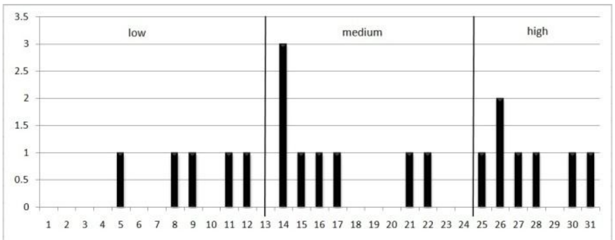

Figure 3. Clustering on frequencies of due date attribute

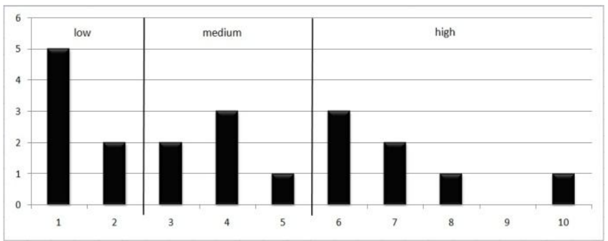

Figure 4. Clustering on frequencies of penalty attribute

3.2. Attribute selection

Attribute-Oriented Induction is a mining method that makes a generalized relation among a task-relevant subset of data, attribute-by-attribute (Cai et al. 1991). Characteristic rules and classification rules are extracted from relational databases by applying hierarchy's concept into an induction procedure. The induction algorithm replaces the low-level concept in a set of variables with its corresponding higher-level concept, and then generalizes the relationship by eliminating identical variables and control the generalization process by using a threshold (Han and Fu 1996).

That is, attribute by attribute, concepts which represent multiple attribute values are replaced with sets of attributes. The set of variables represent rules that describe the data in final relation. A hierarchy's concept outlines a sequence from a set of concepts to their higher-level correspondences. Representing necessary background knowledge by the hierarchies' concept are keys to the generalization process in attribute-oriented induction. They are usually partially ordered according to a general-to-specific ordering (Han and Fu 1996). The discovered rules can be represented in terms of generalized concepts by using hierarchies' concept which users define.

According to the defined technique it is time to study the various attributes which state the solution aspects and the problem condition in order to find the best fitted attributes. Several attributes are considered, and based on the attribute oriented induction structure, finally "processing time", "tardiness penalty" and "due date" are selected as those who describe the thorough aspects of the problem.

The notable point is that Koonce and Tsai (2000) and Kumar and Rao (2009) partitioned the area of each attribute into the equal ranges. This methodology can act well when the attributes values distribute uniformly in its range. But in real world, characteristic values of problems can distribute not uniformly.

To solve this problem, validity index and clustering methods are implemented in this article. The former determined the proper number of clusters and the latter proposed the best possible clustering based on predefined cluster numbers.

3.3. Clustering and validity index

One of the common undirected Data Mining (DM) techniques is clustering, which used wherever there is not any sense about the data treatment and the major target is to classifying the data. In this technique, some patterns are used as indicators to evaluate farness and closeness of the data. The data gathered in a group have the maximum similarity and the data located in different clusters have the maximum dissimilarity with each other about the mentioned pattern.

Table 2. Quantitative to qualitative transformation of attribute values

Job	Processing Time	Due Date	Penalty		Processing Time	Due Date	Penalty
1	5	14	1	→	low	medium	low
2	9	11	2	→	medium	soon	low
3	16	30	1	→	high	late	low
4	7	17	4	→	low	medium	medium
5	23	31	6	→	high	late	high
6	11	28	1	→	medium	late	low
7	1	5	7	→	very low	soon	high
8	6	27	3	→	low	late	medium
9	8	16	5	→	medium	medium	medium
10	15	25	3	→	high	medium	medium
11	19	26	6	→	high	late	high
12	2	15	4	→	very low	medium	medium
13	4	8	8	→	low	soon	high
14	4	14	6	→	low	medium	high
15	9	21	10	→	medium	medium	high
16	9	14	4	→	medium	medium	medium
17	2	12	7	→	very low	soon	high
18	14	22	2	→	high	medium	low
19	15	26	1	→	high	late	low
20	5	9	1	→	low	soon	low

The aim of clustering techniques is to change a partition matrix *U(X)* of a given data set *X* (consisting of *n* patterns, $X = \{x_1, x_2 \dots x_n\}$) so as to find a number, say *K*, of clusters $(C_1, C_2 \dots C_K)$. The partition matrix *U(X)* of size $K \times n$ may be depicted as $U = [u_{kj}]$, $k = 1 \dots K$ and $j = 1 \dots n$, where u_{kj} is the membership of pattern x_j to clusters C_k. In crisp partitioning of the data, the following condition holds: $u_{kj} = 1$ if $x_j \in C_k$, otherwise $u_{kj} = 0$. The purpose is to classify data set *X* such that:

$$C_i \neq \emptyset \qquad i = 1, \dots, k$$

$$C_i \cap C_j = \emptyset \qquad i = 1, \dots, k \quad , \quad j = 1, \dots, k \quad , \quad i \neq j$$

and, $$\cup_{i=1}^{k} C_i = X$$

The k-means algorithm (Pakhira et al. 2004) is one of the well-known clustering methods that produces the minimum squared error partitions. When the number of clusters is known a priori, the k-means algorithm optimizes the distance criterion either by minimizing the within cluster spread, or by maximizing the inter cluster separation

The actual number of clusters and the quality of the clusters are two basic points that should be considered in any clustering algorithm. That is, whatever may be the clustering technique, one has to determine the number of clusters and also the validity of the clusters formed (Pakhira et al. 2004).

The original mission of the validity index algorithms are also answering to these questions precisely. In this algorithm, a series of data firstly collected as the algorithm entry, afterwards exploring under the data is applied to find the best number of clusters.

Table 3. Extracted rules

Rule No.	Processing Time	Due Date	Penalty		Priority			
					1	2	3	4
1	very low	soon	high	→	0.7432	0.2204	0.0364	
2	very low	medium	medium	→	0.0324	0.2372	0.5781	0.1523
3	low	soon	high	→	0.8365	0.1422	0.0213	
4	low	medium	high	→	0.6921	0.2663	0.0416	
5	low	soon	medium	→	0.0902	0.4582	0.4113	0.0403
6	low	medium	medium	→	0.3752	0.5475	0.0773	
7	low	soon	low	→	0.3665	0.5762	0.0573	
8	low	medium	low	→		0.0099	0.3627	0.6274
9	medium	medium	medium	→	0.9187	0.0813		
10	medium	medium	high	→		0.2438	0.6817	0.0745
11	medium	soon	low	→		0.1546	0.8114	0.0340
12	medium	late	low	→			0.2649	0.7351
13	high	medium	medium	→		0.1178	0.4720	0.4102
14	high	late	high	→		0.2114	0.4185	0.3701
15	high	medium	low	→		0.0319	0.3148	0.6533
16	high	late	low	→				1.000

After sufficient iterations, the result is a figure which can show the objective function of the algorithm and as a conclusion the proper number of clusters. According to different types of validity index algorithms, the effective objective function could also be varied.

The mentioned validity index in this article is PBM, proposed in 2004 by Pakhira et al. (2004). This algorithm by the aid of a maximization objective function is looking for proper number of cluster to solve the problem. The PBM validity index could be defined as follows:

$$PBM(X) = \left[\left(1/_{k}\right) \times \left(E_1/_{E_K}\right) \times D_k\right]^2 \quad (5)$$

s.t.

$$E_K = \sum_{k=1}^{K} E_k \quad (6)$$

$$E_k = \sum_{j=1}^{n} u_{kj}\left\|X_j - Z_k\right\| \quad (7)$$

$$D_K = max_{1\le i,j\le K}\left\|Z_i - Z_j\right\| \quad (8)$$

Where K is equal to cluster numbers, E_1 indicates the total distance of the data from the center of cluster in the case that all the data located in just one cluster. E_k is the total data distance from the center of k^{th} cluster if the number of clusters is determined to be K, and D_K, measuring the maximum separation between a pair of clusters. X_j represent the j^{th} input and Z_k is center of k^{th} cluster, finally, U_{kj} is a binary variable which is equal to one if j^{th} input is assigned to the kth cluster.

In the following figures, the results of mentioned algorithm on the sample problem are illustrated, in Figure 2; the attribute of operation time is divided into very low, low, medium and high, in Figure 3, the due date attribute is partitioned as soon, medium and late and the penalty attribute is divided into three types of low, medium and high in Figure 4.

3.4. Transformation of attribute values

In this step, after proper attribute selection and clustering based on validity index output, the results could be applied to the sample problem and transfer the attribute values from quantitative to qualitative, the results are available in table 2.

3.5. Rule Extraction

The designed PSO algorithm was run 1000 times to solve the instance problem generated in section 3.1 and 187 unique solution sequences extracted. The attribute-oriented induction produced 16 distinct characteristic rules for the instance problem (Table 3). This section presents these rules and their application. According to this valuable information and by the aid of attribute characteristics for different jobs, practical rules could be used to decode the procedure for problem solving of PSO algorithm.

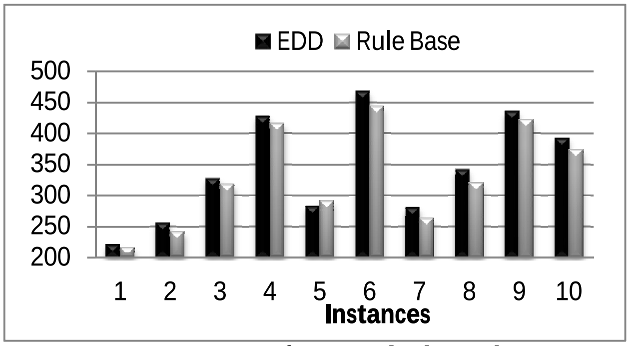

Figure 5. Comparison of extracted rules with EDD

Table 4. Comparison between PSO, Rules and EDD

Instance problems	PSO Min	PSO Ave	Rule Base	EDD	% *
1	205	208.9	**216**	221	2.26
2	226	230.4	**242**	257	5.84
3	302	309.6	**318**	328	3.05
4	399	401.9	**416**	428	2.80
5	264	268.3	291	**283**	**-2.83**
6	415	422.4	**444**	468	5.13
7	239	241.3	**263**	281	6.41
8	292	299.1	**320**	342	6.43
9	405	408.4	**422**	436	3.21
10	349	353.2	**374**	393	4.83

**Improvement*

Clearly, to make a solution for n jobs problem an array with n cells is require. It is considerable that in the proposed algorithm with 20 jobs, solution positions are divided into four part as priority 1 (1, 2, 3, 4, 5), priority 2 (6, 7, 8, 9, 10), priority 3 (11, 12, 13, 14, 15) and priority 4 (16, 17, 18, 19, 20). The probability of a priority class in the rule is defined as the ratio of the frequencies of that class by the total frequencies of all of the classes covered by a rule. That is,

$$Probability\ (p_i) = \frac{f(p_i \in R_a)}{\sum_{i=1}^{4} f(p_i \in R_a)} \qquad (9)$$

in which,

p_i: *is the* i^{th} *priority*

R_a: *is the* a^{th} *rule*

$f(p_i)$: *is the frequencies of* p_i

Then, each job according to its characteristics is assigned to one of the mentioned priorities. Also, jobs in a specific priority are sorted based on the descending order of their presence probability.

In order to assign the jobs to machines, procedure started from the first job in the list (array) and the jobs are assigned to the machine with least machine load.

4. Computational result

In this section, in order to represent the performance of the proposed algorithm, it will be examined and compared with the designed PSO meta-heuristic and the common dispatching rule (Gupta and Sivakumar 2005); Earliest Due Date (EDD) algorithm. So, 10 instance problems are generated by Fisher (1976) standard method and after applying the extracted rules, the results compared with mentioned algorithms.

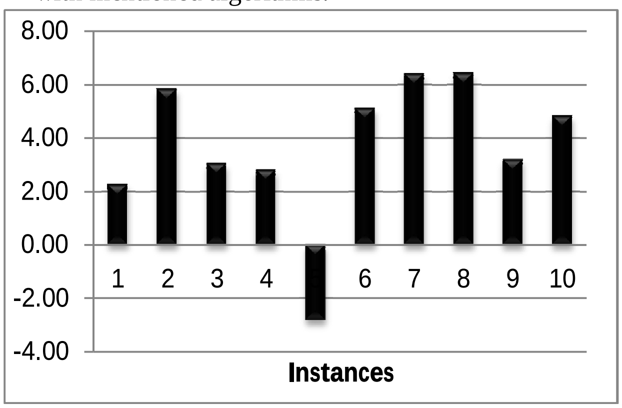

Figure 6. Improvement percentage of extracted rules

Clearly, from the table 4 it could be concluded that, however the proposed algorithm could not defeat the meta-heuristic algorithm, it could hammered the heuristic one. Figure 5, illustrate that in nine out of ten cases, the extracted rules were able to produce shorter objective function of total tardiness than EDD. Moreover, improvement percentages of extracted rules in comparison with EDD on the instances are available in both table 4 and Figure 6 for more perception. The improvement of the algorithm is about 3.7 in average.

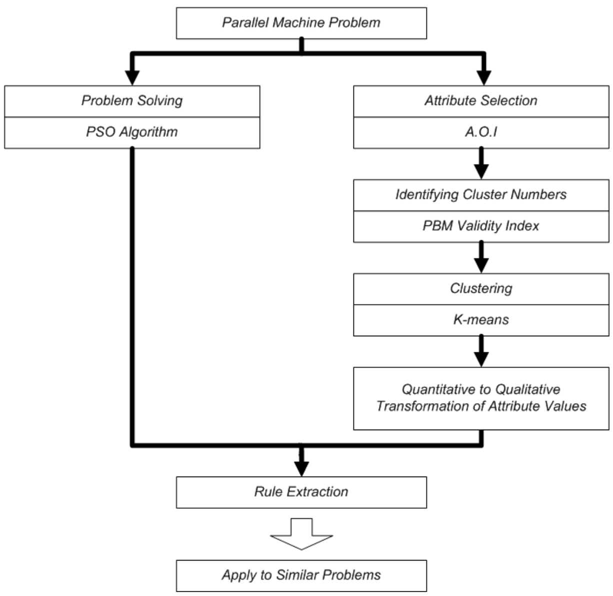

Figure 7. Schematic of proposed algorithm

5. Conclusion

Since scheduling problems are classified in the area of *NP*-hard problems, meta-heuristic algorithms become a common way to solve them, but as the nature of these algorithms, they do not explain how a solution is generated and the characteristics of a proper solution cannot be induced.

The aim of this paper is to fill the gap between mentioned algorithms and the logic of solution making. A parallel scheduling machine with the objective of total tardiness minimization is first solved by PSO meta-heuristic, and then by the aid of A.O.I. technique several attributes are studied to be the parallel machine illustrator. As a result of the data mining technique three attributes, as processing time, due date and penalty, are mentioned. In order to efficient division in each attribute, K-means clustering based on PBM validity index is applied on the instance problem's extracted attributes. Finally the attributes value are transferred from the quantitative status to quantitative, also, 16 rules are extracted by a weighted probability function based on PSO results.

To present the validity of the rules, 10 new instances are generated by Fisher standard method and solved by proposed algorithm in compare with common EDD heuristic and designed PSO. However, the extracted rules were unable to match the performance of the designed PSO; the rules were able to completely outperform the EDD heuristic.

The proposed algorithm initially disclose the black box of the PSO procedure to reach the solutions, also, similar problems could be solved by the extracted rules in analytical way instead of computerize algorithm.

It is recommended to the future researchers to study these kinds of problems to find any unseen attributes and also develop the area of problem into even other scheduling problems. Different data mining techniques also could be examined for future research in order to promote the algorithm.

6. Acknowledgement

The fourth author would like to acknowledge the financial support of University of Tehran for this research under grant number 8109003/1/05.

Corresponding Author:

Dr. Majid Aminnayeri
Department of Industrial Engineering
Amirkabir University of Technology, Tehran, Iran

References

1. Baker, K. R. Introduction to sequencing and scheduling, New York: Wiley, 1974.
2. Cai, Y., Cercone, N. and Han, J. Attribute-oriented Induction in relational databases. Knowledge Discovery in Databases, Cambridge, MA: MIT Press, 1991.
3. Casillas, J., Francisco, J. and Martnez-Lopez Mining uncertain data with 487 multi objective genetic fuzzy systems to be applied in consumer behavior 488 modeling. Expert Systems with Applications, 2009, 36 (2): 1645–1659.
4. Fisher, M. L. A dual algorithm for the one-machine scheduling problem. Mathematical Programming, 1976, 11: 229–251.
5. Gupta, A. K. and Sivakumar, A. L. Single machine scheduling with multiple objectives in semiconductor manufacturing. The International Journal of Advanced Manufacturing Technology, 2005, 26: 950-958.
6. Han, J. and Fu, Y. Attribute-oriented induction in data mining. Advances in Knowledge Discovery and Data Mining, Cambridge, MA: MIT Press, 1996.
7. Kashan, A. H. and Karimi, B. A discrete particle swarm optimization algorithm for scheduling parallel machines. Computer and Industrial Engineering, 2009, 56 (1): 216–223.
8. Khademi Zare, H. and Fakhrzad, M. B. Solving flexible flow-shop problem with a hybrid genetic algorithm and data mining: A fuzzy approach. Expert Systems with Applications, 2011, 38: 7609–7615.
9. Koonce, D.A. and Tsai, S. C. Using data mining to find patterns in genetic algorithm solutions to a job shop schedule. Computers & Industrial Engineering, 2000, 38: 361-374.

10. Kumar, S. and Rao, C. S. P. Application of ant colony, genetic algorithm and data mining-based techniques for scheduling. Robotics and Computer-Integrated Manufacturing, 2009, 25: 901-908.
11. Kusiak, A. and Smith, M. Data mining in design of products and production systems. Annual Reviews in Control, 2007, 31: 147–156.
12. Lee, W. C., Wu, C. C. and Chen, P. A simulated annealing approach to makespan minimization on identical parallel machines. International Journal of Advanced Manufacturing Technology, 2006, 31: 328–334.
13. Niu, Q., Zhou, T. and Wang, L. A hybrid particle swarm optimization for parallel machine total tardiness scheduling. The International Journal of Advanced Manufacturing Technology, 2010, 49: 723-739.
14. Olafsson, S. and Li, X. Learning effective new single machine dispatching rules from optimal scheduling data. International Journal of Production Economics, 2010, 128 (1): 118-126.
15. Ozturk, A., Kayalıgil, S. and Nur, E. O. Manufacturing lead time estimation using data mining. European Journal of Operational Research, 2006, 173: 683–700.
16. Pakhira, M. K., Bandyopadhyay, S. and Maulik U. Validity index for crisp and fuzzy clusters. Pattern Recognition, 2004, 37: 487 – 501.
17. Poursalik K., Miri-Nargesi S. Two Robust Meta-Heuristic Approaches for a New Modeling of Single Machine Scheduling Problem with Multiple Criteria, Journal of American Science, 2011, 7(7): 818-825.
18. Pratchayaborirak, T. and Kachitvichyanukul, V. A two-stage PSO algorithm for job shop scheduling problem. International Journal of Management Science and Engineering Management, 2011, 6(2): 84-93.
19. Sha, D. Y. and Liu, C. H. Using Data Mining for Due Date Assignment in a Dynamic Job Shop Environment. International Journal of Advance Manufacturing Technology, 2005, 25: 1164–1174.
20. Sivrikaya-Serifoglu, F. and Ulusoy, G. Parallel machine scheduling with earliness and tardiness penalties. Computer and Operation Research, 1999, 26 (8): 773–787.
21. Sun, H. and Wang, G. Parallel machine earliness and tardiness scheduling with proportional weights. Computers & Operations Research, 2003, 30: 801–8.
22. Udomsakdigool, A. and Khachitvichyanukul, V. Ant colony algorithm for multi-criteria job shop scheduling to minimize makespan, mean flow time and mean tardiness. International Journal of Management Science and Engineering Management, 2011, 6(2): 117-123.
23. Weckman, G. R., Ganduri, C. V. and Koonce, D. A. A neural network job-shop scheduler. Journal of Intelligent Manufacturing, 2008, 19:191–201.
24. Yang, H. L. and Wang, C.-S. Two stages of case-based reasoning – Integrating genetic algorithm with data mining mechanism. Expert Systems with Applications, 2008, 35 (1–2): 262–272.

12/5/2011