Proceedings of the World Congress on Engineering and Computer Science 2009 Vol II
WCECS 2009, October 20-22, 2009, San Francisco, USA

A New Approach to Generate Rules in Genetic Algorithm Solution to a Job Shop Schedule by Fuzzy Clustering

S. Nima Rafinejad, Faraz Ramtin, Alireza Boloori Arabani

Abstract— In this article, data mining methodologies are used to explore the rules in data generated by a genetic algorithm performing a job shop scheduling operation. In each solution of the genetic algorithm, every individually scheduled operation of a job is treated as a decision which contains knowledge. Each decision is modeled as a function of a set of job characteristics (e.g., remaining processing time), which are divided into classes. In the literature, job characteristics are divided into classes arbitrary without any reasoning. But it should mention that in real problems, job characteristics may not distributed uniformly in their ranges. Fuzzy c-mean clustering is used to divide job characteristics into classes in logical manner. Finally, results are compared by DOE techniques and show that using this method can significantly improve the objective function of the scheduling problem.

Index Terms— **Job shop scheduling, Genetic algorithm, Fuzzy clustering, Attribute-oriented induction, Data mining**

I. INTRODUCTION

Job-shop scheduling problem is one of the well-known hardest combinatorial optimization problems [1]. In job shop scheduling problem, we are given a set of jobs and a set of machines. Each machine can handle, at most, one job at a time. Each job consists of a chain of operations, each of which needs to be processed during an uninterrupted time period of a given length on a given machine. The purpose is to find a schedule, that is, an allocation of the operations to time intervals on the machines, which has a minimum duration required to complete all jobs.

During the last three decades, this problem has captured the interest of a significant number of researchers. A lot of literature has been published, but no efficient solution algorithm has been found yet for solving it to optimality in polynomial time. This has led to recent interest in using heuristic algorithms to address the problem [2].

With the rapid increase in the speed of computing and the growing need for efficiency in scheduling area, it becomes increasingly important to explore ways of obtaining better schedules at some extra computational costs. The genetic algorithm approach is one such kind of attempt. In the view of computational cost, genetic algorithms are not as efficient as some other heuristic methods [3]. Genetic algorithms (GAs) constitute a branch of evolutionary computation (EC). They are well-known for the merit of global exploration. Following the law of 'survival-of-the-fittest', GAs encode solutions into chromosomes (individuals) and evolve them by executing iterative genetic operators [4]. Genetic algorithms often provide fast solutions to traditional numeric problems. For example, a GA can generate schedules for a manufacturing job shop. However, GAs does not demonstrate repeatability or provide an explanation of how a solution is developed.

Using data mining, can induce rules from the solutions of a GA, which describe its behavior. Data mining can be used to learn from job shop schedules produced by genetic algorithms. In practice, the effort required to duplicate the GA's performance was significant.

Attribute-Oriented Induction, as one of the data mining methods, is a set-oriented database mining method that generalizes a task-relevant subset of data, attribute-by-attribute, into a generalized relation [5].

This method was developed to extract characteristic rules and classification rules from relational databases by employing concept hierarchies into an induction process [6]. In this article, using data mining to find patterns in genetic algorithm solutions to a job shop schedule, is considered which is done by Koonce and Tsai [6], but this paper has some novel contribution, which can make the performance measure of job shop problem closer to the optimal. In this part, fuzzy c-mean clustering is used to divide job characteristic into classes.

Cluster analysis target is to cluster a data set into subgroups of similar characteristics. The conventional (hard) clustering methods make each part of the data set belong to exactly one cluster. Since fuzzy theory was created by Zadeh [7] which produced the idea of allowing to data to have membership values in each cluster, fuzzy clustering has been widely applied in a variety of cases [8]. In fuzzy clustering, a single point can have partial membership in more than one class [9].

The authors are with the department of Industrial Engineering, Amirkabir University of Technology, 424 Hafez Avenue, Tehran, Iran (Alireza Boloori Arabani is the corresponding author, phone: +98-21-77633696, +98-912-5498037; e-mail: alirezaboloori@aut.ac.ir, alirezaboloori@yahoo.com).

II. METHODOLOGY

The goal of the current study is to develop a data mining algorithm with a new fuzzy clustering approach for a single job shop problem.

A. Data collection

At first, a database needs to be identified for the job shop problems with the aim of making general rules and training process. The knowledge base for the learning duty is produced from a job shop problem which has an optimal solution generated by a GA. A well-known 6 × 6 problem instance, first proposed by Muth and Thompson [10] has been selected as the benchmark problem. This problem has six jobs, each with six operations to be scheduled on six machines and has an optimal solution of 55 units for the performance measure of makespan. The data for the instance is shown in Table I which the first figure is the machine number and the second figure is the processing time.

B. Genetic construction

A genetic algorithm should construct for the single job shop problem. In this article, with the aim of comparing the result with Koonce and Tsai [6], all the operator in genetic algorithm is designed same as that paper. This research uses Syswerda's [10] list of ordered operations representation as the gene model for a job shop schedule. The 6×6 job shop is encoded into a 36 integer array. For example, a solution might be the sequence: {2, 6, 5, 2, 6, 1, 5, 4, 6, 3, 4, 1, 3, 6, 2, 4, 5, 4, 2, 4, 3, 3, 1, 3, 1, 6, 5, 5, 1, 3, 4, 2, 6, 5, 1, 2}. Each element in the array corresponds to a job. Successive references to a job in the array imply the next available operation for that job.

Table I
Machine order and processing time

Job	Operation					
	1	2	3	4	5	6
1	3,1	1,3	2,6	4,7	6,3	5,6
2	2,8	3,5	5,10	6,10	1,10	4,4
3	3,5	4,4	6,8	1,9	2,1	5,7
4	2,5	1,5	3,5	4,3	5,8	6,9
5	3,9	2,3	5,5	6,4	1,3	4,1
6	2,3	4,3	6,9	1,10	5,4	3,1

The population size was set to 500 and an initial population was created by random selection. The selection scheme keeps the half of population with better fitness (shorter makespan). The 250 selected solutions (chromosomes) were used to produce 500 schedules, which were joined to the initial population to produce 1000 strings. From those 1000 solutions, 500 strings with best fitness will move to next generation.

The method of crossover used was OC2 (Order-based Crossover, [11]). For each mating pair, a group of random gene sites was selected for the parent strings substitute, the probability of a site being chosen was 60%. Following crossover, all child genes were mentioned to order-based mutation. Using a mutation probability of 20%, each chromosome chosen for mutation had two genes for mutation chosen at random.

C. Data mining

Before data mining duties, the algorithm must be determined and the data structured for mining properly. For this task, attribute-oriented induction was selected as the mining algorithm and the data was prepared accordingly.

Attribute-oriented induction employing concept hierarchies into an induction process to extract characteristic rules and classification rules from relational databases. The induction algorithm exchanges the low-level concept in a tuple with its corresponding higher-level concept, and then generalizes the relationship by eliminating identical tuples and using a threshold to control the generalization process [12]. That is, attribute by attribute, concepts which represent multiple attribute values are substituted for sets of attributes. The tuples in final relation represent rules that describe the data.

The following classification hierarchies are based on Koonce and Tsai [6] for the 6×6 job shop problem, but it should mention that other sizes of job shop problems may require other classification approaches.

The aim of this paper is to find relationship between an operation's characteristics and its order in the GA solution sequence. That is, it wants to predict the sequence position of an operation by giving its characteristics.
The attribute classification is as following:

C. 1. PRIORITY

The attribute Priority is determined as a range of sequence positions in the solution. Thus, the value of position is classified into one of six classes: {1; 2; 3; 4; 5 or 6} as 1, {7; 8; 9; 10; 11 or 12} as 2, {13; 14; 15; 16; 17 or 18} as 3, {19; 20; 21; 22; 23 or 24} as 4, {25; 26; 27; 28; 29 or 30} as 5, and {31; 32; 33; 34; 35 or 36} as 6.

C. 2. OPERATION

The Operation attributes represent the sequence of the operation in the job which is an ordinal variable. It was decided that four classes of operation consider for induction. Operation 1 was classified as “first”, the next two operations as “middle”, operations 4 and 5 as “later”, and operation 6 was classified as “last”. In this paper these four classes of operation, “first”, “middle”, “later” and “last” denote as “1”, ”2”, ”3” and ”4”, respectively in attribute rules tables.

C. 3. OTHER ATTRIBUTE

In Koonce and Tsai [6] for processing time attribute, remaining processing time attribute and machine loading attribute, levels of attributes are considered arbitrary. For example processing time attribute, the classes are obtained by dividing the range of processing time into three equal parts.

This methodology can act well when the attribute values distribute uniformly in its range. But in real world, characteristic values of problems can distribute not uniformly. In this paper, first fuzzy *c*-mean clustering is used to cluster characteristic values into proper parts. Second a new method proposed to determine the suitable boundary between clusters area for each attribute. On the other side, attribute-oriented induction method extracts qualitative rules, so fuzzy clustering approach can be suitable. As mentioned in fuzzy *c*-mean clustering, the center of clusters and the membership values of each data to each cluster are gained. These membership values were applied to determine the boundary between clusters. When the data are divided into suitable number of clusters, by using membership value of each data in each cluster, these boundaries obtain as follow:

$$\text{Boundary}_{12} = \frac{v_1 \sum_{j=1}^{nd} \mu_{1j} + v_2 \sum_{j=1}^{nd} \mu_{2j}}{\sum_{j=1}^{nd} \mu_{1j} + \sum_{j=1}^{nd} \mu_{2j}} \quad (1)$$

where:

i index of cluster

j index of data

nd number of the data

v_i *i*th cluster center

μ_{ij} membership value of *j*th data in *i*th cluster

The boundary is determined by the weighted average of distance between two cluster centers, that is, the sum of membership values of data in each cluster is used as the weight for that cluster center. This method augments the concept of membership value to determine the boundary between clusters.

In this case, processing time attribute is divided into properly number of three clusters which denote as “1”, “2” and “3” in attribute rules tables, remaining processing time attribute is divided into three clusters which denote as “1”, “2” and “3” in attribute rules tables and machine loading attribute is divided into two clusters which denote as “1”, and “2” in attribute rules tables. According to equation (1), boundary between two clusters is obtained.

III. Case study

As mentioned before, this article focuses on a well-known 6 × 6 problem instance, first proposed by Muth and Thompson [10]. At first, processing time, remaining processing time and machine loading data are divided into mentioned number of clusters by fuzzy *c*-mean clustering. Then, by the means of equation (1), new boundaries between attribute levels are determined. Considering these new boundaries, 36 rules are extracted from 36 operations of sample problem, 24 distinct rules are found. The priority probabilities of each rule are gained by using the technique of Koonce and Tsai [6]. In this case, new rule base priorities probabilities are trained by 1000 optimal solutions, that is obtained by solving the sample problem by genetic algorithm. The rule base and its probabilities are shown in Table II and Table III, respectively.

Table II
Attribute Rules

Attribute			
Operation	Processing Time	Remaining Processing Time	Machine Loading
1	1	3	1
1	2	3	1
1	3	2	1
1	3	3	1
2	1	2	1
2	1	3	1
2	1	3	2
2	2	1	2
2	2	2	1
2	2	3	1
2	2	3	2
2	3	2	2
2	3	3	2
3	1	1	1
3	1	1	2
3	1	2	1
3	2	1	2
3	3	1	2
3	3	2	1
3	3	2	2
4	1	1	1
4	2	1	1
4	2	1	2
4	3	1	2

The rule base above does not cover the entire range of possible attribute combinations. A simple probability based induction gives a set of rules that assign a priority based on the operation's place in a job. Table IV and Table V give the set of generalized rules and their probabilities, respectively that will be used to schedule operations that fail to fire any induced rules.

Table III
Attribute Rules' Probabilities

Priority					
1	2	3	4	5	6
58.7	34.35	6	0.95		
83.8	13.5	2.25	0.45		
31.3	59	9.7			
99.9	0.1				
2.2	35.2	42.2	17.9	2.5	
15.2	66.8	17.5	0.5		
11.1	38.7	35.3	13.3	1.6	
	6.3	29.4	45.7	18	0.6
0.2	19.15	43.85	29.05	7.7	0.05
54.65	29.6	11.85	3.6	0.3	
27.3	54.5	16.8	1.4		
2.4	35.25	44.25	14.95	3.1	0.05
13.4	49.8	30.8	5.9	0.1	
	0.2	6.1	28.5	44	21.2
		0.1	5.8	49.3	44.8
	4.3	36.3	48	11.4	
	0.35	6.05	30.1	50.5	13
0.034	5.066	27.23	42.67	23.1	1.9
	0.3	4.5	39	50.5	5.7
	2.55	29.35	35.85	27.95	4.3
	0.05	0.6	4.55	16.25	78.55
		1.1	15.3	46.3	37.3
			0.2	10.4	89.4
			2.85	17.85	79.3

Table IV
Generalized Attribute Rules

Attribute			
Operation	Processing Time	Remaining Processing Time	Machine Loading
1	Any	Any	Any
2	Any	Any	Any
3	Any	Any	Any
4	Any	Any	Any

Table V
Generalized Attribute Rules' Probabilities

Priority					
1	2	3	4	5	6
70.2	26.77	3.03			
14.9	36.17	37.25	11.4	0.28	
	0.44	11.24	38.01	40.08	10.23
			11.9	19.26	79.55

IV. Computational Result

The applicability of these 24 rules was tested using 76, 6 ×6 job shop test cases generated not uniformly. These cases were all scheduled using the GA, Koonce and Tsai [6] method and propose method. Table VI shows that in the test problems which the characteristic values are not distributed uniformly in its range, the result of proposed method significantly dominates Koonce and Tsai [6] method. It can be seen by performing ANOVA test on 76 problem samples, by which the significance of our applied measure is emphasized. These measures are as *F*-Value=10.52 and *P*-Value=0.00146. Furthermore, 10 problem samples (illustrated in Table VI) are picked from those 76 problem samples described before.

Table VI
Table of Results

Problem	GA	Proposed Method	Koonce & Tsai
1	223	262	330
2	217	271	331
3	201	201	255
4	192	233	279
5	207	293	338
6	207	253	295
7	196	236	277
8	212	270	308
9	209	265	302
10	217	295	330

V. Conclusion

Data mining can be used to extract knowledge from a job shop schedules solved by genetic algorithms.

Genetic algorithms provide schedules for a job shop scheduling. However, GAs do not demonstrate process of how a solution is developed. This paper presents a method for extracting rules from the solutions of a GA, which describe its behavior by using data mining attribute-oriented induction technique.

Attribute-oriented induction determines four attributes for this job shop problem and partitioning each attribute values range by using arbitrary boundaries. In real world, characteristic values of problems can distribute not uniformly in its range. So this method may not work properly for this kind of problems. In this paper, fuzzy c-mean clustering is used to cluster characteristic values into proper parts. Then a new method proposed to determine the suitable boundary between clusters area for each attribute.

After extracting a set of 24 rules, which, combined with 4 default rules learned from the problem data, this method is applied to problems with the same structure (6×6 Job Shop) and different operation times and sequences, the rules were able to consistently outperform the Koonce and Tsai. However, the learned rules were unable to match the performance of the genetic algorithm on these problems. Future research should incorporate incremental learning into the mining process an allowing for multiple schedule scenarios in the data sets.

REFERENCES

[1] M. Garey, D. Johnson and R. Sethi.. "The complexity of flowshop and jobshop scheduling". Maths Ops Res, vol. 1, pp. 117-129, 1976.
[2] R. Cheng, M. Gen and Y. Tsujimura. "A tutorial survey of job-shop scheduling problems using genetic algorithms, part II: representation". International Journal of Computers and Industrial Engineering, vol. 30, pp. 983-97, 1996.
[3] E. Nowicki, C. Smutnicki." A fast taboo search algorithm for the job-shop problem". Management Sciences, vol. 42, no. 6, pp. 797-813, 1996.
[4] D. E. Goldberg. "Genetic Algorithms in Search Optimization and Machine Learning", Addison-Wesley: Massachusetts, 1989.
[5] Y. Cai, N. Cercone and J. Han. "Attribute-oriented Induction in relational databases, Knowledge Discovery in Databases". Cambridge, MA: MIT Press, 1991.
[6] 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, no. 3, pp. 361–374, 2000.
[7] L. A. Zadeh. "Fuzzy sets". Inform. Control vol. 8, pp. 338–353, 1965.
[8] J. C. Bezdek. "Pattern Recognition with Fuzzy Objective Function Algorithms". Plenum Press, New York, 1981.
[9] T. J. Ross. "Fuzzy logic with engineering applications". New York: McGraw-Hill, 1995.
[10] J. F. Muth and G. L. Thompson. "Industrial scheduling", Englewood Cliffs, NJ: Prentice Hall, 1963.
[11] G. Syswerda. "Schedule optimization using genetic algorithms, Handbook of Genetic Algorithms". New York, NY: Van Nostrand Reinhold, 1991.
[12] J. Han and Y. Fu. "Attribute-oriented induction in data mining, Advances in Knowledge Discovery and Data Mining". Cambridge, MA: MIT Press, 1996.