Send Orders for Reprints to reprints@benthamscience.ae

The Open Electrical & Electronic Engineering Journal, 2014, *8*, 653-657 653

Open Access

The Dynamic Job Shop Scheduling Approach Based on Data-Driven Genetic Algorithm

Yanfang Yu*[1] and Yue Ying[2]

Open and Distance Education Research Institute, Zhejiang Radio & Television University, Hangzhou 310030, P.R. China

Abstract. Rapid development of the Internet of Things not only provides large amounts of data to the job-shop scheduling, but also proposes a great challenge for dynamic job shop scheduling. A dynamic job shop scheduling approach is proposed based on the data-driven genetic algorithm. Application examples suggest that this approach is correct, feasible and available. This approach can provide the technical support for the long-term development of enterprises in the field of intelligent production.

Keywords: Data-driven, dynamic optimization, genetic algorithm, job shop scheduling.

1. INTRODUCTION

With rapid development of modern science and technology, especially the information technology (IT), an agile manufacturing system with characteristics of digitization, information and intelligence has appeared. For a dynamic, changeable agile manufacturing system, a dynamic job shop scheduling approach is one of effective measures for production management [1]. The typical characteristic of dynamic scheduling is ensuring high efficiency of real-time production and flexibility of perturbation response [2]. The foundation and the key to realize advanced manufacture and improve production effectiveness are to explore efficient dynamic scheduling approaches [3].

In recent years, more and more scholars have begun to focus on and study the agile manufacturing system oriented job shop scheduling and dynamic scheduling [4-8]. Chao-Yong Zhang et al. investigated the dynamic scheduling in the cases of delayed arrival, processing of raw materials and assembling, as well as adding the new jobs urgently [9]. Brank et al. studied the dynamic scheduling in the case of random arrival of jobs, and improved the scheduling performance through combining the early idle time [10]. Liu et al. shortened the idle time earned from production process by the job selection rule to improve the scheduling performance [11]. Adibi studied how to handle the dynamic events - the random arrival of jobs and machine error, and assessed the scheduling performance through minimizing the processing time and the job delay [12]. Considering adding new machines, Fattahi investigated a dynamic flexible job shop scheduling problem (JSSP) with changeable arrival time of new jobs and processing time, and measured the effectiveness of scheduling by minimizing time of completion [13].

With continuous upgrade and a wide application of internet of things, mass data of device, process and production reflecting running state of the agile manufacturing system have been collected and stored [14-16]. However, in practical optimization process, it is usually lack of an efficient statistic, analysis and evaluation of these data, so they cannot be converted to useful information to management departments. The data-driven optimization method is a new approach to study the agile manufacturing system.

2. DATA-DRIVEN GENETIC ALGORITHM

The Genetic algorithm derives from biogenetics and the natural law of survival of the fittest. It is an intelligent optimization method with an iterative process of “survival + detection”. Genetic algorithms express problem solving as the survival of the fittest of chromosomes, and converge to an individual the most adaptable to the environment finally by step iteration of chromosomes. In the optimization process of existing genetic algorithms, it often lacks an effective statistic, analysis and application of off-line and on-line data of optimization systems. In view of this, this paper attempts to design and implement a data-driven genetic algorithm for the dynamic JSSP. And its basic framework is as shown in Fig. (**1**).

In this paper, the data-driven genetic algorithm is defined as a hybrid genetic algorithm combining the genetic optimization model and the knowledge model. In this algorithm, the genetic optimization model searches the feasible space of the optimization problem according to the “neighborhood search” strategy. While the knowledge model mines useful scheduling knowledge from mass off-line and on-line data, and then uses it to guide the follow-up optimization processes. The data-driven genetic algorithm is based on the genetic optimization model, and meanwhile, highlights the role of the knowledge model. It optimizes both models to improve the algorithm optimization efficiency.

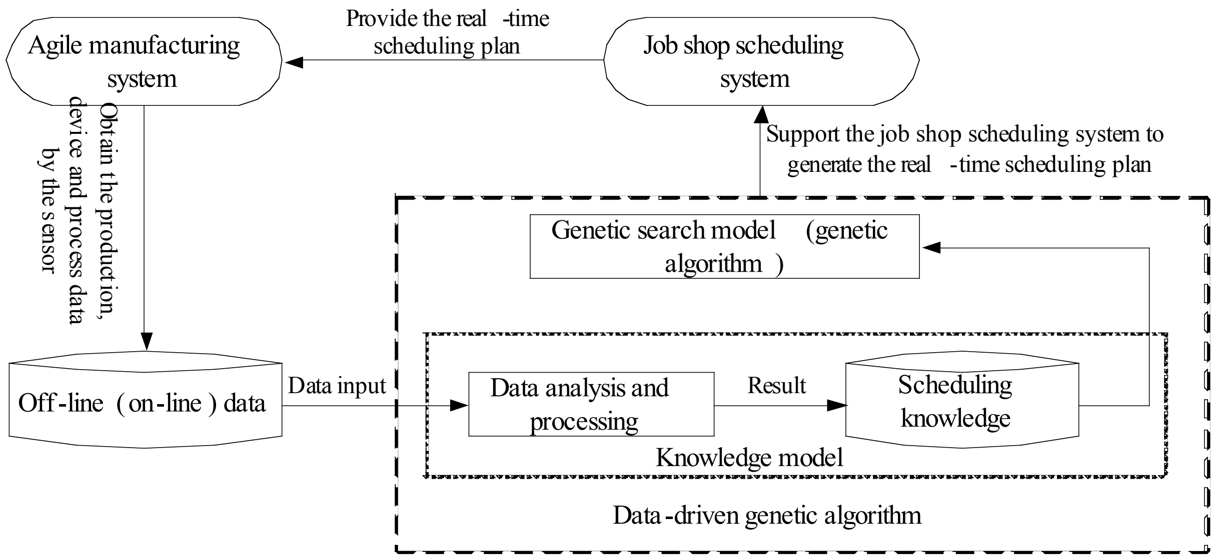

Fig. (1). The basic framework of data-driven genetic algorithm.

1.1. Definition of Scheduling Knowledge

The dynamic job shop scheduling usually needs to consider chance events like rush order, equipment failure and job re-doing. After a chance event happens, it certainly will disturb execution of the original scheduling plan and the plan must be rescheduled. In order to implement rescheduling quickly and effectively, useful scheduling knowledge needs to be mined from mass off-line and on-line data for guiding the follow-up rescheduling processes.

(1) Parameter knowledge. Usually, the genetic algorithms are very sensitive to parameters. If parameter selection is unreasonable, then it is often very hard to converge to the quasi-optimal solution. Oppositely, it usually converges to the quasi-optimal solution quickly. The problem of how to properly set the parameters of genetic algorithms is always a hot issue for scholars at home and abroad. In order to reduce the sensitivity of genetic algorithms to parameters, this paper employs multiple sets of typical parameters to implement the evolutionary process of genetic algorithms, meanwhile, determines parameter combinations to be used for next iteration according to the work performance of each set of parameters. After the genetic algorithm completes the iteration, if the global optimal solution has been improved, then call the current iteration as a success. In the process of solving current examples of job shop scheduling, the iterations with success obtained by a given parameter combination should be the work performance of the combination. The parameter knowledge refers to a kind of accumulated knowledge of work performance of parameter combinations, represented as:

$$K_{Para} = \{PA_1, PA_2, \cdots, PA_s\} \tag{1}$$

In which, KPara denotes the array of work parameters of different parameter combinations, PAi denotes the work performance of the ith set of parameters, S is the number of parameter combinations.

(2) Operator knowledge. For the dynamic JSSP, it is very difficult to find out a general operator able to solve various examples effectively. For some scheduling examples, a better result could be obtained by such a scheduling operator. While for some others, the other scheduling operators could be more effective. Therefore, it is very necessary to study the applicability of genetic operators to different examples of the dynamic JSSP. In order to solve this problem effectively, the author integrated several different genetic operators into the proposed approach, and mined some operators able to solve current examples effectively based on mass off-line and online data. Assume that the current crossover operation is performed by a crossover operator OX1, and the set of two parent individuals before the crossover operation is PopB, then the set of two offspring individuals after the crossover operation is PopA. If the best individual in PopA is better than that in PopB, then this crossover operation is successful; otherwise, unsuccessful. In the process of solving the current scheduling example, the iterations with success obtained by a given operator should be the work performance of this operator. The operator knowledge refers to a kind of accumulated knowledge of work performance of operators, represented as:

$$K_{Oper} = \{OP_1, OP_2, \cdots, OP_s\} \tag{2}$$

In which, KOper denotes the array of work performance of different operators, OPi denotes the work performance of the ith operator, S is the number of operators.

(3) Strategy knowledge. For the dynamic JSSP, it is usually not easy to obtain the quasi-optimal solution purely by the genetic algorithm. For that reason, it is very necessary to integrate the user (expert) experience (knowledge) into the algorithm as a strategy. The scheduling strategy can be summarized as priority processing the tasks of a higher priority, the shortest and the longest processing time. For different dynamic scheduling examples, we should keep trying and make dynamic regulation to determine which scheduling strategy can obtain the quasi-optimal solution. In order to effectively solve different examples, this paper employs multiple scheduling strategies to implement the evolutionary process of genetic algorithms, and meanwhile, determines

the strategy for the next iteration according to the work performance of each scheduling strategy. In the process of solving the current example, the iterations with success obtained by a given strategy should be the work performance of the scheduling strategy. The strategy knowledge refers to a kind of accumulated knowledge of work performance of scheduling strategies, represented as:

$$K_{Str} = \{ST_1, ST_2, \cdots, ST_s\} \tag{3}$$

In which, KStr denotes the array of work performance of different scheduling strategies, STi denotes the work performance of the set of parameters, S is the number of parameter combinations.

1.2. Guidance for Genetic Algorithms Based on Scheduling Knowledge

With the continuous running of the production scheduling system, the off-line and on-line data of device, process and production reflecting the running state have been accumulated constantly. This paper mined the knowledge of parameter, operator and strategy from these off-line and on-line data, and then used it to guide the follow-up optimization processes of genetic algorithms.

(1) The three-stage division for the job shop scheduling plan. When solving the dynamic JSSP, the whole solving process should be planned reasonably to adapt the uncertainty and dynamics of the scheduling system. Considering the above, this paper divided the job shop scheduling plan into three stages: the executed plan, the plan to be executed and the one to be programmed (as shown in Fig. (**2**)). The executed plan represents the scheduling plans executed by the scheduling system until now. And for such plans, they won't be changed basically. The plan to be executed refers to the scheduling plans that will be processed in the scheduling system. They have some dynamics and certainty, needing to be handled immediately by some selfreparing operations. The plan to be programmed refers to the scheduling plans that will be processed in the scheduling system after a longer follow-up time. They have some dynamics and uncertainty, needing to be handled immediately by some genetic algorithms. In the proposed algorithm, the strategy knowledge is used to guide the selfreparing operations, and the operator knowledge and parameter knowledge to guide the genetic optimization procedure. It is important to note that the reasonable division of the plan to be executed and the one to be programmed is usually determined by the historical data, user experience and expert knowledge. And this paper will divide these two plans based on the historical data.

(2) The selfreparing operation based on strategy knowledge. When solving the dynamic JSSP, this paper mainly employs the following three scheduling strategies: give priority processing to the task with a higher priority (Str1), the one with the shortest processing time (Str2) and the one with the longest processing time (Str3). Here suppose that the work performance of each scheduling strategy is ST1, ST2 and ST3 respectively (at the initial phase, their work performance is initialized to 1). When implementing the selfreparing operation, this paper selects a scheduling strategy based on the following probability:

$$PR_i = \frac{ST_i}{\sum_{k=1}^{3} ST_k} \tag{4}$$

In which, PRi denotes the selective probability of the scheduling strategy.

(3) The genetic optimization operation based on parameter knowledge and operator knowledge. At the initial phase of data-driven genetic algorithm, the author used the orthogonal design method to generate 27 typical parameter combinations, and meanwhile, initialized the work performance of each set of parameters to 1. Before each iterative, the data-driven genetic algorithm adopts the roulette method to randomly select a set of parameters from 27 sets as the parameters for this iteration based on the work performance of parameter combinations. And the compute mode of selective probability of each set of parameters is as below:

$$PR_i = \frac{PA_i}{\sum_{k=1}^{27} PA_k} \tag{5}$$

In which, PRi denotes the selective probability of the ith parameter combination; PAi is the work performance of the ith set of parameters.

In the data-driven genetic algorithm for the dynamic JSSP, the author employed five types of crossover operators and three types of mutation operators to implement its optimization process together. Here, suppose that the work performance of each crossover operator is OPXi (*i*=1, 2,…, 5) and that of each mutation operator is OPMi (*i*=1, 2, 3) respectively. Before each iterative, this algorithm will select a

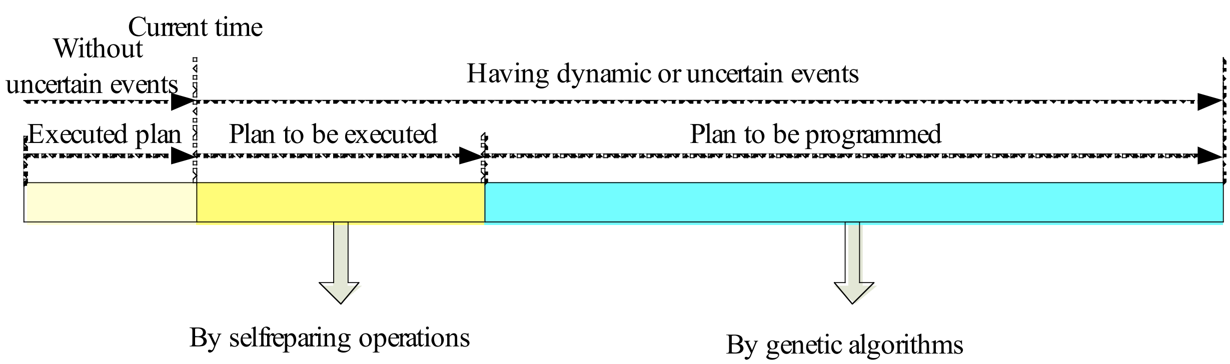

Fig. (2). The three-stage division for the scheduling plan.

kind of crossover operator and mutation operator with the following probabilities respectively.

$$PRX_i = \frac{OPX_i}{\sum_{k=1}^{5} OPX_k} \tag{6}$$

$$PRM_i = \frac{OPM_i}{\sum_{k=1}^{3} OPM_k} \tag{7}$$

2. EXPERIMENTAL RESULTS

As shown in Table **1**, five instances are used to test the feasibility and correctness of the proposed algorithm.

In order to simulate the uncertainty and dynamics in the real job shop scheduling system, this paper adopts the following way to add some uncertainty and dynamics to the above instances:

(1) Randomly add 5% new processes, and their processing time on the jth machine is λTij. Here, λ is a random number in an interval [0.8, 1.2], Tij is the average processing time for all the processes of the ith task on the jth machine.

(2) Randomly change the processing time of 5% existing processes, which is changed to λTij. Here, λ is a random number in an interval [0.8, 1.2], Tij is the processing time for the current process of the ith task on the jth machine.

The author employed the Visual C++ language design and implemented the data-driven genetic algorithm for the dynamic JSSP. All experiments were completed in a desk computer with a CPU of Pentium Dual Core 2G and 2G memory. In order to assess the optimization performance of the proposed algorithm more objectively, this paper solved each instance 100 times by this genetic algorithm and the average value of experimental results was taken as the final result. The experimental results are shown in Table **2**.

It can be seen from Table **2**, although some uncertainty and dynamics were added to these five instances respectively, the proposed algorithm can solve these uncertain events effectively. For the variable quantity of total processing time, that of the instance MK07 is minimum (6.28%), MK10 is maximum (11.05%), and their average value is 8.91%. For the variable quantity of process processing plan, that of MK07 is minimum (5.51%), MK09 is maximum (8.15%), and their average value is 6.98%. So from the above results, it is not hard to see whether the variable quantity of total processing time or that of process processing plan, the data-driven genetic algorithm can deal with the above dynamics and uncertainty by a smaller variable quantity (less than 10%). In general, the application examples have shown the correctness, feasibility and usability of the proposed algorithm.

CONCLUSION

In the modern dynamic job shop scheduling system, mass production, device and process data have been collected and stored. However, in the real optimization process, they cannot be converted into useful information for management departments. In view of this, a dynamic scheduling method based on the data-driven genetic algorithm was proposed. And the application examples have shown the correctness, feasibility and usability of the proposed algorithm. The data-driven optimization method created a new way to study the agile manufacturing system.

Table 1. Five instances of job shop scheduling.

Name of instance	Number of jobs	Number of machines	Sum of processes
MK06	10	10	150
MK07	20	5	100
MK08	20	10	225
MK09	20	10	240
MK10	20	13	240

Table 2. Experimental results of the five instances solved by the proposed algorithm.

Name of instance	Sum of processes	Variable quantity of Total processing time (%)	Variable quantity of Process processing plan (%)
MK06	150	8.13	6.72
MK07	100	6.28	5.51
MK08	225	9.27	7.32
MK09	240	11.05	8.15
MK10	240	9.82	7.18

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Supported by the social science research project of Zhejiang province (2013Z11) and the science and technology project of Zhejiang science and Technology Bureau (2014C31068).

REFERENCES

[1] H. G. Xiong, J. J. Li, J. Y. Kong, G. Z. Jiang, and J. T. Yang, "Research on Integration of Job Shop Scheduling and Monitor System in One-of-a-kind Production System", Manufacture *Information Engineering of China*, vol. 35, no. 1, pp. 21-27, 2006.

[2] Q. K. Pan and J. Y. Zhu, "Rolling Time Horizon Job-Shop Scheduling Strategy in Dynamic Environment", *Journal of Nanjing University of Aeronautics & Astronautics*, vol. 37, no. 2, pp. 262-268, 2005.

[3] X. L. Wu, "Research on Flexible Job Shop Dynamic Scheduling Problem", *Journal of System Simulation*, vol. 20, no. 14, pp. 3828-3832, 2008.

[4] M. Gholami and M. Zandieh, "Integrating simulation and genetic algorithm to schedule a dynamic flexible job shop", *Journal of Intelligent Manufacturing*, vol. 20, no. 4, pp. 481-498, 2009.

[5] Z. L. Liu, "Evaluation on developing level of unban agglomeration derived from resources exploration", *Journal of Applied Sciences*, vol. 13, no. 21, pp. 4702-4707, 2013.

[6] C. Chen and L. N. Xing, "GA-ACO for solving flexible job shop scheduling problem", *Computer Integrated Manufacturing Systems*, vol. 17, no. 3, pp. 615-621, 2011.

[7] D. Bhattacharya and S. Roychowdhury, "A Constrained Cost Minimizing Redundancy Allocation Problem in Coherent Systems with Non-overlapping Subsystems", *Advances in Industrial Engineering and Management*, vol. 3 no. 3, pp. 1-6, 2014. doi:10.7508/AIEM-V3-N3-1-6

[8] C. Y. Zhang, X. Y. Li, X. J. Wang, Q. Liu, and L. Gao, "Multi-objective dynamic scheduling optimization strategy based on rolling-horizon procedure", *China Mechanical Engineering*, vol. 20, no. 18, pp. 2190-2197, 2009.

[9] J. Brank and D. Mattfeld, "Anticipation and flexibility in dynamic scheduling", *International Journal of Production Research*, vol. 43, no. 15, pp. 3103-3129, 2005.

[10] M. Z. Liu, H. Shan, and Z. Q. Jiang, "Dynamic rescheduling optimization of job-shop under uncertain conditions", *Journal of Mechanical Engineering*, vol. 45, no. 10, pp. 137-142, 2009.

[11] M. Adibi, M. Zandieh, M. Amiri, "Multi-objective scheduling of dynamic job shop using variable neighbourhood search", *Expert Systems with Applications*, vol. 37, no. 1, pp. 281-287, 2010.

[12] P. Fattahi and A. Fallhi, "Dynamic scheduling in flexible job shop systems by considering simultaneously efficiency and stability", *Journal of Manufacturing Science and Technology*, vol. 2, no. 2, pp. 114-123, 2010.

[13] J. MacGregor and A. Cinarc, "Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods", *Computers and Chemical Engineering*, vol. 47, pp. 111-120, 2012.

[14] B. Wei, "Research on Dynamic Assessment and Optimization of Vehicle Scheduling with Data Driven", M.S. thesis, Beijing Jiaotong University, 2011

[15] L. Xiong, "Data-driven technology based process monitoring and optimization", M.S. thesis, Zhejiang University, 2008.

Received: September 18, 2014 Revised: December 22, 2014 Accepted: December 31, 2014