IEMS Vol. 6, No. 2, pp. 119-124, December 2007.

A Taguchi Approach to Parameter Setting in a Genetic Algorithm for General Job Shop Scheduling Problem

Ji Ung Sun †
School of Industrial & Managment Engineering
Hankuk University of Foreign Studies Kyongki-do 449-791, KOREA
Tel: +82-31-330-4191, E-mail: jusun@hufs.ac.kr

Received Date, December 2006; Accepted Date, April 2007

Abstract. The most difficult and time-intensive issue in the successful implementation of genetic algorithms is to find good parameter setting, one of the most popular subjects of current research in genetic algorithms. In this study, we present a new efficient experimental design method for parameter optimization in a genetic algorithm for general job shop scheduling problem using the Taguchi method. Four genetic parameters including the population size, the crossover rate, the mutation rate, and the stopping condition are treated as design factors. For the performance characteristic, makespan is adopted. The number of jobs, the number of operations required to be processed in each job, and the number of machines are considered as noise factors in generating various job shop environments. A robust design experiment with inner and outer orthogonal arrays is conducted by computer simulation, and the optimal parameter setting is presented which consists of a combination of the level of each design factor. The validity of the optimal parameter setting is investigated by comparing its SN ratios with those obtained by an experiment with full factorial designs.

Keywords: Genetic Algorithm, Parameter Optimization, Taguchi Method, Robust Design, General Job Shop Scheduling.

1. INTRODUCTION

Recently, the application of heuristic search method for combinatorial optimization problems has attracted much attention. Since Holland (1975) developed the basics of genetic algorithm (GA), it has been proven to be efficient for many complicated problems. The most difficult and time-intensive issue in the successful implementation of GA is to find good parameter setting, one of the most popular subjects of current research in GA. Recently, a number of approaches have been suggested to derive robust parameter settings for GA. In one of the most extensive studies for determining the optimal parameter values, Schaffer *et al*. (1989) concluded that the optimal parameter settings vary from problem to problem. Pakath and Zaveri (1993) proposed a decision support system to determine the appropriate parameter values systematically for a given problem. Gupta *et al*. (1993) discussed an experimental design approach using full factorial design to evaluate parameter settings. Hsieh *et al*. (2000) used the Taguchi method to find the optimal operating parameters in the GA such that the efficiency of the GA can be improved. More recently, nevertheless, attention has been shifted toward the processes of forming new trial chromosomes at each interation in the GA such that the efficiency of the GA can be further improved, as was done in the works of Leung and Wang (2001) and Liu *et al*. (2006). More specifically, for instance, can one use the abilities of the Taguchi method to seek the optimal breeding to efficiently generate optimal offspring in establishing an algorithm of even higher performance.

In this study, we present a new efficient experimental design method for parameter optimization in a GA for general job shop scheduling problem (GJSP) using Taguchi method. Robust designs such as Taguchi method borrow many ideas from the statistical design of experiments for evaluating and implementing improvements in products, processes, or equipment. Its fundamental principle, largely speaking, is to improve the quality of characteristic of interest by minimizing the effect of the causes of variation, but not eliminating those causes themselves. To do so, four most commonly studied GA parameters including the population size, the crossover

† : Corresponding Author

rate, the mutation rate, and the stopping condition that implies the number of iterations which an improvement is not occurred consecutively are treated as design factors. For the performance characteristic, makespan is adopted. The number of jobs, the number of operations

required to be processed in each job, and the number of machines are considered as noise factors in generating various job shop environments. An experimental design is constructed using inner and outer orthogonal arrays for the design and noise factors, respectively. At each combination of design and noise factors, the performance characteristic is obtainded, and the so called signal-to-noise (SN) ratio is calculated at each design condition as a robust measure. These SN ratios are analyzed by analysis of variance technique to determine optimal settings of design factors that are robust to noise factors. The validity of the optimal parameter settings is investigated by comparing its SN ratios with those obtained by an experiment with full factorial designs.

This paper is organized as follows. Taguchi's philosophy for performance improvement is reviewed in section 2. In section 3, the job shop scheduling problem is described and experimental factors chosen for this study are presented. The design of an orthogonal array experiment and the analysis of the experimental data are presented in section 4. And conclusions are found in section 5.

2. TAGUCHI METHOD

In this section, we present the basic concept of the Taguchi method. Performance of a production process, as measured by some performance characteristics, vary due to a variety of causes. We call all such causes noise factors. The fundamental principle of robust design (or parameter design) proposed by Taguchi is to determine settings of design factors such that the effects of noise factors on the performance characteristics are minimized (Phadke 1989). For this purpose, Taguchi suggests an experimental approach in which orthogonal arrays are used to determine a large number of design factors with a small number of experimental runs. He also proposed to capture the effects of noise factors on a performance characteristic by the SN ratio.

2.1 Overview and Experimental Strategy

For robust design, Taguchi suggests to employ an experiment in which orthogonal arrays are used for the arrangement of design and noise factors. The design factors are assigned to the inner array and noise factors to the outer array. Noise factors are not mixed with the design factors in a single orthogonal array. Instead, noise factors are arranged separately to form different testing conditions so that the sensitivity of a performance characteristic to noise factors can be measured by an appropriate SN ratio. It is important in every parameter design work to identify important noise factors. Engineering experiences and judgments are needed in identifying them. Figure 1 shows a typical experimental plan for parameter design.

	Inner array	Outer array	
	design factors D1 D2 D3	1 2 2 1 1 2 1 2 1 1 2 2	N3 N2 noise N1 factors
1 2 3 4	1 1 1 1 2 2 2 1 2 2 2 1	Experimental Data y_{ij}	SN Ratios SN_i

Figure 1. Experimental strategy for parameter design

2.2 Loss Functions and SN Ratios

Taguchi classifies performance characteristic (y) into three categories, i.e., the-smaller-the-better (SB) the-larger-the-better (LB), and a-specific-target-best (TB) cases. For instance, makespan is the-smaller-the-better type, and throughput is of the-larger-the-better type. Taguchi suggests that performance be measured by the loss incurred due to the deviation of the performance characteristic from its target value. In general, the exact functional form of the loss may be unknown or complex, and therefore, Taguchi recommends to use a quadratic loss function. For example, let y be TB performance characteristic and t be its target value. Then, the quadratic loss function is defined as $L(y) = k(y-t)^2$. Similarly, for SB and LB cases the quadratic loss functions are respectively given by $L(y) = ky^2, L(y) = k/y^2$. The expected loss has two components: one related to the deviation of the mean y from the target value and the other is the variance of y due to noise factors. The objective of a robust design is to find the setting of process design factors that minimize the expected quadratic loss, which is equi-valent to maximize the SN ratio. Depending on the nature of y, Taguchi suggests different SN ratios to be maximized.

3. GJSP MODEL AND GENETIC ALGORITHM

3.1 Problem Description of GJSP Model

The manufacturing system under study consists of several machines. A set of jobs j_i, i = 1, 2, ..., n, is available for processing at time zero. Job j_i requires n_i number of operations with ik and P_{ik}, k = 1, 2, ... , n_i, known. Some jobs require processing on certain machines more than once in their operation sequences with reentrant work flows. Some machines require a setup prior to processing each job where setup times are sequence dependent.

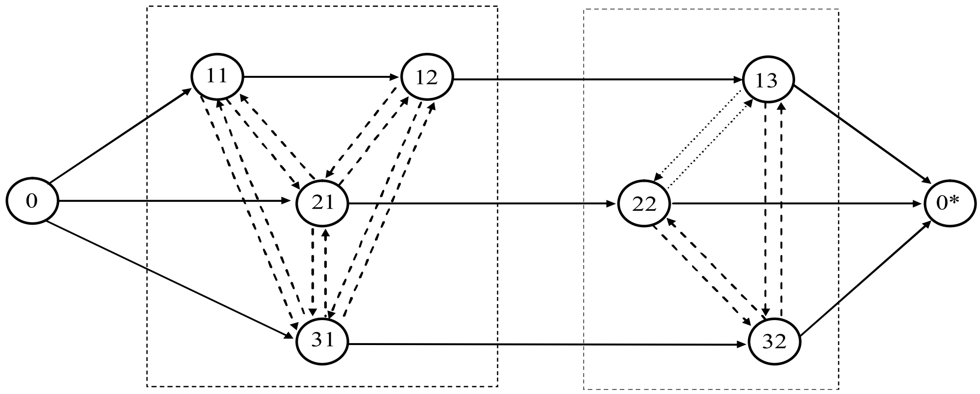

Figure 2. DGR with three jobs and two machines

It is useful to represent this problem on a disjunctive graph $F(N, A, E)$, with node set N, conjunctive arc set A, and disjunctive arc set E. The nodes of N correspond to operations, the directed arcs of A to precedence constraints among the operations of the same job, and the pairs of disjunctive arcs of E to pairs of operations to be performed on the same machine (Balas 1969). The JSP under study can be represented by using the disjunctive graph model (DGM) presented by Ovacik and Uzsoy (1997). An example of the DGM of a problem with two machines and three jobs is shown in Figure 2.

Operations 11, 12, 21 and 31 are processed on the first machine while 13, 22 and 32 are processed on the second machine. Node that source node *0* and sink node *0** denotes the beginning of processing in the system and the completion of the last job, respectively.

3.2 GA for GJSP Model

For scheduling problems, Chen *et al*. (1995) presented application of GA to flow shop problems in order to minimize makespan. Lee and Choi (1995) developed a GA for a single machine scheduling problem with due dates constraints. Cheng *et al*. (1996, 1999) gave a tutorial survey of job shop scheduling problems using GA. Koonce and Tsai (2000) presented a novel use of data mining algorithms to explore the patterns in data generated by a GA for job shop scheduling. Liu et al. (2006) proposed the hybrid Taguchi-genetic algorithm which the Taguchi method is inserted between crossover and mutaion operations of a GA to solve the classical job shop scheduling problem.

3.2.1 Chromosomal representation

The first step of developing GA is to encode a solution as a finite-length string called chromosome. Representation plays a key role in the development of GA. For each machine in the GJSP under study, some operations have the precedence constraints due to reentrant work flows. Thus it is important to develop a encoding method without violating the feasibility of a chromosome. In this study, a solution is encoded as a $|M|$ (denote the number of machines) dimensional vector in which each element is a permutation of job numbers for each machine and decoded as a $|M|$ dimensional vector in which each element is a permutation of operation numbers. Therefore, the length of the chromosome equals to the total number of operations to be sequenced.

For example, consider the problem with two machines and three jobs shown in figure 1. Note that the operation 11, 12 and 13 are element of job 1, 21 and 22 are job 2, and operation 31 and 32 are job 3. Assign a number, from 1 to 7, to each operation in the order of job indices, we have, 1 = 11, 2 = 12, 3 = 13, 4 = 21, 5 = 22, 6 = 31, and 7 = 32. Note that operation 1, 2, 4 and 6 are performed on the first machine and operation 3, 5 and 7 are performed on the second machine. And also note that there is a precedence constraint between operation 1 and 2 on the first machine. Then a chromosome can be represented as a two dimensional vector such as [(1 3 2 1), (2 1 3)] by generating two 1s, one 2, and one 3 for the first machine and one 1, 2 and 3 for the second machine where 1 represents job number 1, 2 job number 2 and 3 job number 3, respectively. The above chromosome can be directly decoded as [(1 6 4 2), (5 3 7)], which implies a complete schedule, by assigning a number, from 1 to 7, to each position starting from job 1 to 3, sequentially. If the decoded solution turns out to be infeasible, the repairing method proposed in section 3.1 is invoked. Job number encoding/operation number decoding method facilitates that the feasibility conditions are basically not violated by the crossover operations.

3.2.2 Reproduction, Crossover and mutation

Three types of genetic operators, namely reproduction, crossover and mutation, are applied to an initial population randomly generated. Chromosomes with higher fitness values are selected for crossover and mutation using the reproduction operator. In this study, stochastic remainder sampling without replacement scheme (Goldberg 1989) is employed to reduce the stochastic errors associated with the traditional roulette wheel selection. In this process, each chromosome receives copies as many as integral parts of expected count whereas fractional parts are treated as success probabilities. One by one, weighted coin tosses (Bernoulli trials) are performed using the fractional parts as success probabilities. This process continues until the size of population reaches a certain value.

After reproduction, members of the newly reproduced chromosomes are mated at random. Thereafter, crossover operator is invoked to produce two offspring. Crossover plays the role of exchanging information among chromosome. In this study, we use the order crossover (OX) operator (Goldberg 1989) which can be applied from the first element to the last ($|M|$ th element) consecutively between the two elements located at the same position in the considered pair of chromosomes. In the OX operator, two chromosomes are aligned, and two crossing sites are picked at random along the elements. These two points define a matching section. To illustrate the OX operator, consider the first two elements of considered pair of chromosomes:

$$c_1 = 1\ 3\ 2\ |\ 1\ 1\ 2\ |\ 3\ 3$$
$$c_2 = 3\ 2\ 1\ |\ 2\ 3\ 1\ |\ 3\ 1$$

Observing that there are two 1s and one 2 in the matching section of c_1, the mate of c_2, we randomly pick the same number of elements among 1s and 2s from c_2. Suppose the numbers marked by an H are picked in c_2 as follows :

$$c_2 = 3\ \mathrm{H}\ \mathrm{H}\ |\ 2\ 3\ \mathrm{H}\ |\ 3\ 1$$

Then we exchange the Hs with the neighborhood num-bers until all the Hs are in the matching section as follows.

$$c_2 = 3\ 2\ 3\ |\ \mathrm{H}\ \mathrm{H}\ \mathrm{H}\ |\ 3\ 1$$

Following the OX operator, offspring c_2' is created.

$$c_2' = 3\ 2\ 3\ |\ 1\ 1\ 2\ |\ 3\ 1$$

In the same way, c_1' can be obtained. With the OX operation, each offspring contains information partially determined by each of its parents. This process continues to the last element consecutively.

Since our coding structure treats a schedule as a permutation vector, mutation is implemented by some type of exchange of position among the jobs. In this study, the following mutation operator is utilized in a encoded chromosome. Given the current chromosome, select an element at random among $|M|$ elements. Within the selected element, select two positions with different values at random and then swap the genes.

3.2.3 Fitness function and replacement strategy

The objective in GA is to find chromosome which gives the maximum fitness function value. In this study, the following fitness function is adopted:

$$\pi_{jt} = \frac{\sum_{j \in C_t} g_{jt}}{g_{jt}} \tag{1}$$

where g_{jt} is the makespan value of c_{jt}, the jth chromosome in generation t.

Also, to preserve the best member's performance in the next generation, the elitist strategy (Goldberg 1989) is adopted. As a result, the best performing chromosome of the previous generation is alive in the current population.

4. EXPERIMENTAL DESIGN AND ANALYSIS

4.1 Design Factors and Their Levels

Once a genetic algorithm is developed, its performance strongly relies on the parameters of GA. In this study, we select four most commonly studied GA parameters, i.e. population size, stopping condition that implies the number of iterations which an improvement is not occurred consecutively, crossover rate, and mutation rate. After an extensive preliminary analysis of the algorithm, we choose three levels for each parameter values. Selected design factors and their levels are listed in Table 1.

Table 1. Design factors and their levels

factor \ level	1	2	3
(1) population size (A)	50	100	150
(2) stopping condition (B)	30	40	50
(3) crossover rate (C)	0.7	0.8	0.9
(4) mutation rate (D)	0.5	0.7	0.9

4.2 Noise Factors and Their Levels

To minimize the sensitivity of a performance characteristic to noise factors, we first need to estimate the sensitivity in a consistent manner for any combination of the design factor levels. This is achieved through a proper selection of testing conditions. Various noise factors exist in a job shop type manufacturing system. Although it is not necessary to include all noise factors, we must use engineering judgment to decide which are more important and what testing conditions are appropriate to capture their effects. In this study, three noise factors are selected while others are judged to be less important and therefore ignored. Selected noise factors and their levels are listed in Table 2.

Table 2. Noise factors and their levels

factor \ level	1	2	3	4
(1) number of jobs (U)	10	20	30	40
(2) number of operations per job (V)	10	15	20	
(3) number of machines (W)	10	15	20	

4.3 Experimental Design

In this study, we want to estimate the main effects of design factors as well as the interactions *AB*, *AC* and *BC*. From the experience, the interactions between the population size and the stopping condition is expected. Interactions between the population size and the crossover rate, and between the stopping condition and the crossover rate are also of interest. An efficient way of studying the effects of several design factors simultaneously is to plan a matrix experiment using an orthogonal array. For the inner array, we choose orthogonal array $L_{27}(3^{13})$ which has 13 three-level columns and 27 rows. Design factors A through D are respectively assigned to columns 1, 2, 5 and 9. Array containing 12 combinations of noise factors are used to determine the test conditions. Factors *U*, *V* and *W* are respectively assigned to columns 1 through 3. Each row of the inner array L_{27} represents a design of the process. Performance (makespan) of each design is evaluated by computer experiment under each noise condition specified by the outer array.

4.4 Data Analysis

The purpose of conducting a matrix experiment is to determine the level of each factor that gives the highest SN ratio. For the present problem, SN ratios at each row of the inner array are calculated as follows. Since the ideal value of the makespan is zero, the corresponding SN ratio is given by

$$\eta = -10\log\left(\frac{1}{12}\sum_{j=1}^{12} y_j^2\right), \tag{2}$$

where y_j is the normalized deviation of the observed makespan at test condition *j* from the minimum value of the condition *j*. Maximizing η leads to the minimization of the quality loss. After the SN ratios are calculated, the next step in data analysis is to evaluate the significance of each effect on the characteristic. Based upon analysis of variance (ANOVA), optimum levels of design factors that are statistically significant at level 0.05 and their contribution ratios are shown in Table 3.

From Table 3, the optimum settings of design factors for η are determined as $A_2B_3D_1$. Factor *C* has no significant effect on the makespan. Therefore, the optimum level of *C* is determined by the level whose value

Table 3. Optimum levels with respect to η

factor	Optimal level (contribution ratio)
A	A_2 (7.62 %)
B	B_3 (23.24 %)
C	
D	D_1 (9.22 %)
AB	A_2B_3 (34.02 %)
AC	
BC	

of η is maximum, C_3. The results of a verification experiments by the full factorial design show that the settings $A_3B_3C_2D_1$ gives the highest η. Also, η at $A_2B_3C_3D_1$ is very close to η at $A_3B_3C_2D_1$. Based on these fact, we may conclude that optimum design $A_2B_3C_3D_1$ is a good compromising solution. Note that this optimal setting is obtained with a far smaller experiment ($L_{27} \times L_{12}$) than full factorial ($L_{81} \times L_{12}$).

5. CONCLUSIONS

Taguchi's parameter design method is an important tool for robust design. Robust design is an engineering methodology for optimizing product and process conditions that are minimally sensitive to the causes of variation, and which produce high-quality products with low development and manufacturing costs.

In this study, we present a new efficient experimental design method for parameter optimization in a genetic algorithm for general job shop scheduling problem using the Taguchi method. Four genetic parameters including the population size, the crossover rate, the mutation rate, and the stopping condition are treated as design factors. The number of jobs, the number of operations required to be processed in each job, and the number of machines are considered as noise factors in generating various job shop environments. A robust design experiment with inner and outer orthogonal arrays is conducted by computer simulation, and the optimal parameter setting is presented which consists of a combination of the level of each design factor. The validity of the optimal parameter setting is investigated by comparing its SN ratios with those obtained by an experiment with full factorial designs.

ACKNOWLEDGMENT

This work was supported by Hankuk University of Foreign Studies Research Fund of 2007.

REFERENCES

Ballas, E. (1969), Machine sequencing via disjunctive graphs: an implicit enumeration approach, *Operations Research*, **17**, 941-957.

Chen, C. L., Vempati, V. S., and Aljaber, N. (1995), An application of genetic algorithms for flow shop problems, *European Journal of Operations Research*, **80**, 389-396.

Cheng, R., Gen, M., and Tsujimura, Y. (1996), A tutorial survey of job-shop scheduling problems using genetic algorithms, part I: representation, *Computers and Industrial Egningeering*, **30**, 983-996.

Cheng, R., Gen, M., and Tsujimura, Y. (1999), A tutorial survey of job-shop scheduling problems using genetic algorithms, part II: hybrid genetic search strategies, *Computers and Industrial Egningeering*, **36**, 343-364.

Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization and Machine Learning. Reading*, Massachusetts: Addison Wesley.

Gupta, M, Gupta, Y., and Kumar, A. (1993), Minimizing flow time variance in single machine system using genetic algorithms, *European Journal of Operations Research*, **70**, 289-303.

Hsieh, C., Chou, J., and Wu, Y. (2000), Taguchi-MHGA method for optimizing grey-fuzzy gain-scheduler, *Proceedings of the 6th International Conference on Automation Technology*, Taiwan, 575-582.

Holland, J. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press.

Koonce, D. A. and Tsai, S. C. (2000), Using data mining to find patterns in genetic agorithm solutions to a job shop schedule, *Computers and Industrial Engineering*, **38**, 361-374.

Lee, C. Y. and Choi, J. Y. (1995), A genetic algorithm for job sequencing problems with distinct due dates and general early-tardy penalty weights, *Computers and Operations Research*, **22**, 857-869.

Leung, Y. and Wang, Y. (2001), An orthogonal genetic algorithm with quantization for global numerical optimization, *IEEE Transactions on Evolutionary Computation*, **5**, 41-53.

Liu, T., Tsai, J., and Chou, J. (2006), Improved genetic algorithm for the job-shop scheduling problem, *International Journal of Advanced Manufacturing Technology*, **27**, 1021-1029.

Ovacik, I. M. and Uzsoy, R. (1997), *Decomposition methods for complex factory scheduling problems*, Norwell, MA: Kluwer Academic.

Pakath, R. and Zaveri, J. S. (1993), *Specifying critical inputs in a genetic-driven decision support system: an automated facility*, Working Paper, University of Kentucky, Lexington.

Phadke, M. S. (1989), *Quality engineering using robust design*, Prentice-Hall International.

Schaffer, J. D., Caruana, R. A., Eshelman, L. J., and Das, R. (1989), A study of control parameter affecting online performance of genetic algorithms for function optimization, *Proceedings of the 3rd International Conference on Genetic Algorithms*, Arlington, VA.

Sun, J. U., Yee, S. R., and Hwang, H. (2003), Job shop scheduling with sequence dependent setup times to minimize makespan, *International Journal of Industrial Engineering-Theory, Applications And Practice*, **10**, 455-461.