Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 589317, 13 pages
http://dx.doi.org/10.1155/2015/589317

Research Article

Applying Genetic Algorithm to Generation of High-Dimensional Item Response Data

ByoungWook Kim,[1] JaMee Kim,[2] and WonGyu Lee[2]

[1] *Department of Computer Science Education, Korea University, Seoul 136-701, Republic of Korea*
[2] *Department of Computer Science and Engineering, Korea University, Seoul 136-701, Republic of Korea*

Correspondence should be addressed to WonGyu Lee; lee@inc.korea.ac.kr

Received 28 November 2014; Accepted 3 February 2015

Academic Editor: Sanghyuk Lee

The item response data is the *nm*-dimensional data based on the responses made by *m* examinees to the questionnaire consisting of *n* items. It is used to estimate the ability of examinees and item parameters in educational evaluation. For estimates to be valid, the simulation input data must reflect reality. This paper presents the effective combination of the genetic algorithm (GA) and Monte Carlo methods for the generation of item response data as simulation input data similar to real data. To this end, we generated four types of item response data using Monte Carlo and the GA and evaluated how similarly the generated item response data represents the real item response data with the item parameters (item difficulty and discrimination). We adopt two types of measurement, which are root mean square error and Kullback-Leibler divergence, for comparison of item parameters between real data and four types of generated data. The results show that applying the GA to initial population generated by Monte Carlo is the most effective in generating item response data that is most similar to real item response data. This study is meaningful in that we found that the GA contributes to the generation of more realistic simulation input data.

1. Introduction

The purpose of computer simulation is to model a certain phenomenon or incident virtually in an attempt to predict the results of a real-life situation. It is both a cost effective and time saving method for testing "what-if" scenarios? [1, 2]. A simulation can be run to conduct a virtual study in order to predict results of a real-life situation and establish various mathematical models (probability distribution) specifically for certain problems. Therefore, it is possible to select the most effective method from among various situations [3]. Simulation is used in cases where mathematical analysis is difficult due to the complexity of the real-world situation or experimentation is simply impossible or too costly due to the restrictions of the real world. That is, simulation is applied to various engineering areas, such as queuing problems [4], inventory management problems [5], network problems such as PERT/CPM [6], problems related to production (production allocation, production integration, production system balance, factory location analysis, etc.) [7–9], repair of machinery and equipment [10], replacement problem [11], treasury loans, and investment planning [12].

In educational evaluation, simulation is used to estimate item parameters and the abilities of examinees [13] because it is difficult to obtain the real item response data for estimating the parameters of items and abilities. Therefore, when a controlled experiment needs to be conducted to obtain statistical results regarding item parameters in educational evaluation, studies are conducted based on item response data generated through simulation [14, 15]. The main advantage of simulation studies is that researchers can define the ability of examinees and characteristics of items according to the research purposes. If they know the characteristics of examinees and items, researchers can control data and manipulate certain factors to evaluate the effects [13]. Until now, when item response data is generated, based on item response theory (hereinafter IRT), Monte Carlo, which uses the probability distribution of examinees correctly answering items depending on their ability, has been used [16].

The key to simulation studies is to model problems in real life as realistically as possible. Also, to guarantee the validity and reliability of simulation results, simulation data is more important than anything else [17, 18]. To evaluate the validity of simulation results, it is necessary to use statistical techniques to compare the results of actual systems and simulation results. However, not enough research has been conducted with regard to evaluating the fitness of the item response data generated through Monte Carlo.

As computation performance improves, computer science technology has been introduced into areas where simulation data is generated. To guarantee high quality in the software industry, testing software is required. To achieve this objective, we need to test software thoroughly with adequate test data. The automatic generation of a test suite and its adequacy are the key issues when testing a software product. Some studies have been conducted to show that the field of software testing used a genetic algorithm (hereinafter GA) to generate automated software test data and improve the performance of tests [19–22].

The GA is a computational model based on the evolutionary process seen in the natural world. It is a global optimization technique developed by John Holland in 1975 and one of the techniques for solving optimization problems. The GA models the evolution of life and the evolutionary mechanism using engineering methods and uses them for solving problems and learning systems.

In the field of educational evaluation, the GA is used for test-sheet composition [23–28]. The test construction problem (test construction problem or item selection problem) can be formulated as a zero-one combinatorial optimization [29]. The running time will increase exponentially with the number of items in the item bank. This problem has been proven to be an NP-hard problem; that is, the solution time is not bound by a polynomial of the size of the problem [30].

The GA is known to be effective in finding the optimal solution for NP-hard problems, and it has been utilized in areas dealing with item data, but not for generating item response data. In other words, studies have not validated item response data so far, or the GA has not been utilized in the field of educational evaluation in some cases. Accordingly, this paper is trying to verify whether the item response data generated for simulation studies are similar to real item response data. Also, the GA is used for proposing a method of generating item response data. To this end, item response data, generated using Monte Carlo and the GA, was compared. Item difficulty and item discrimination, representing the characteristics of the item response data of real examinees, were used to generate item response data. To evaluate how similarly the generated item response data represents the real item response data model, the item difficulty and discrimination of the generated item response data were compared with those of the real item response data.

2. Background

2.1. Test Theory. Item response data is that which shows whether the examinees responded correctly or incorrectly to the items making up a test sheet. Based on test theory, item response data is used to estimate item characteristics and the ability of examinees. According to test theory, tests are analyzed indirectly by measuring the latent trait of people (specifically the test takers) and the items making up the test [31]. As item analysis must precede test analysis, test theory generically refers to not only items, but also the tests and all related theories. Test theory is divided into the classical test theory and item response theory. This paper deals with item response data, with item characteristics used to evaluate the fitness of the GA. Accordingly, the methods of obtaining item difficulty and item discrimination in the classical test theory and item response theory are examined.

2.1.1. Classical Test Theory. According to classical test theory, analysis is conducted based on the total score of the test tools, with the assumption that the observed score of the test is composed of the true score and error score. Also, as the true score of examinees cannot be known, the mean of the scores, obtained by infinitely repeating theoretically identical tests for the same examinees, is used to presume the true score. Item difficulty and item difficulty according to classical test theory are as follows.

(1) Item Difficulty. Item difficulty is an index indicating the degree to which an item is deemed easy or difficult. It is the ratio, that is, probability, of examinees who answer correctly to total examinees. The item difficulty for item i, P_i, is defined as the proportion of examinees who get that item correct. The formula for calculating item difficulty is shown below:

$$P = \frac{R}{N}, \tag{1}$$

where P is the difficulty of a certain item, R is the number of examinees who get that item correct, and N is the total number of examinees.

(2) Item Discrimination. Item discrimination is the index indicating the degree to which an item discriminates examinees depending on abilities. If high-ability examinees correctly answered an item and low-ability examinees failed to answer an item correctly, this item is regarded as an item that functions properly. In other words, if the score of those examinees who correctly answered an item is high and the score of those examinees who got the wrong answer for the same item is low, this item can be said to be an item able to discriminate examinees. In contrast, if the score of those examinees who correctly answered the item is low and the score of those examinees who got the wrong answer for the item is high, this item can be said to be an item with negative discrimination. Also, if those who correctly answered an item get the same score as those examinees who got the wrong answer for the item, this item will be an item that has no discrimination, that is, an item whose discrimination index is 0. Therefore, the discrimination index of an item is estimated by the correlation coefficient between the item score and

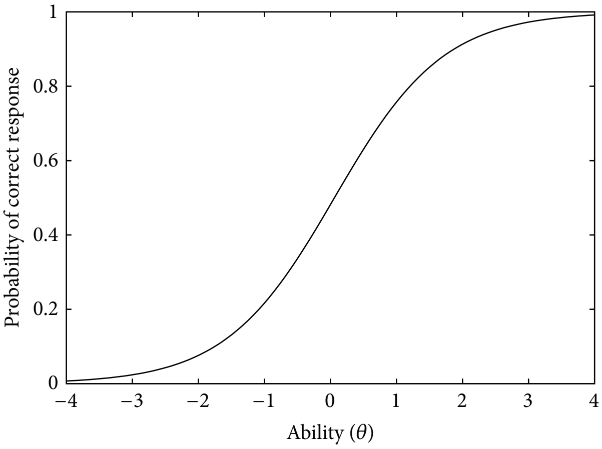


Figure 1: Item characteristic curve.

the total score of examinees. The formula for the correlation coefficient that estimates item discrimination is shown below:

$$r = \frac{N\sum XY - \sum X \sum Y}{\sqrt{N\sum X^2 - \left(\sum X\right)^2}\sqrt{N\sum Y^2 - \left(\sum Y\right)^2}}, \qquad (2)$$

where N is the total number of examinees, X is the item score of each examinee, and Y is the total score of each examinee.

2.1.2. Item Response Theory. Item response theory does not analyze an item based on the total test score, but, as each has an invariable unique trait, it is a test theory that analyzes the item based on the item characteristic curve (hereinafter ICC) indicating this attribute. Therefore, in item response theory, one of the most important concepts is the item characteristic curve. The ICC is a curve indicating the probability of correctly answering an item as shown in Figure 1.

In Figure 1 the horizontal axis is the ability, a potential characteristic of a person, designated as θ. The person's ability ranges between almost none and infinitely high, but it was standardized and converted into a score with the mean being zero (0) and the standard deviation being one (1). Therefore, 95% of all examinees are between -1.96θ and $+1.96\theta$ for the ability range. In general, when the item characteristic curve is drawn, the horizontal axis, which indicates ability, ranges between −3 and +3 because the ability of almost all the examinees is in this range. The vertical axis indicates the probability of correctly answering an item depending on ability and ranges between 0.0 and 1.0.

Item response data is used for simulation studies based on IRT. IRT is a test theory for measuring the characteristics (θ) of examinees, which consists of the difficulty and discrimination of evaluation items based on the responses to the evaluation items. The key characteristic of IRT is that, when parameters, such as personal ability and item difficulty, are calculated, it approaches individual evaluation items probabilistically with discrete results like correct answers. That is, mathematical models are applied to test data [31].

In the IRT model θ is used to indicate the estimated latent trait capability of examinees as measured by test items. A variety of IRT models have been developed. This paper is based on two parameters. According to IRT, the higher the ability (θ) of examinees, the higher the probability of correctly answering an item. In the two-parameter logistic IRT model, the probability of examinees correctly answering an item is defined as follows:

$$P_i\left(\theta_j\right) = \frac{1}{1 + e^{-Da_i(\theta_j - b_i)}}, \qquad (3)$$

where e is mathematical constant, i is the number of items ($i = 1, 2, 3, \ldots, n$), j is examinees ($j = 1, 2, 3, \ldots, n$), and a_i is discrimination of item i. At $\theta_j = b_i$, it is proportionate to the slope of the item response function. b_i is the difficulty of item i. θ_j is ability of examinee j, $P_i(\theta_j)$ is the probability that examinee j, who has θ_j ability, will respond correctly to item i, and D is a scaling factor of 1.702.

2.2. Monte Carlo Method. The Monte Carlo method is a concept contrary to deterministic algorithms. It is a sort of randomized algorithm that uses random numbers to calculate the value of a function [32]. As the algorithm includes many iterative operations and calculations, Monte Carlo is suitable for computer calculations. Monte Carlo method is one of the techniques for randomly selecting (sampling) the values for simulation from a probability distribution. It is also known as a simulated sampling technique. The advantage of Monte Carlo method is that it generates a random number for a condition corresponding to the input, examines all possible cases, and supports decision-making by providing the distribution and statistics resulting from the output.

Accordingly, Monte Carlo method is a method of approximating the desired solution or law by using random numbers to create data and synthesizing the manipulated results of sufficient numbers or random experiments when a certain problem is given. In the field of educational evaluation, Monte Carlo method is also used to generate item response data of examinees [33].

2.3. Genetic Algorithm

2.3.1. The Genetic Algorithm Process. The GA is one of the techniques used for probabilistic investigation, learning, and optimization. It is based on two salient theories of genetics. One is Charles Darwin's theory of survival of the fittest; that is, those individuals who adapt well to nature will survive, and those who do not will die out, and the other is Mendel's law; that is, the traits of descendants are inherited from the genes received from both parents [34]. The genes in the chromosomes of higher forms of life use crossover and mutation to evolve to an optimal state with each passing generation. This GA was created as part of an effort to use this law to search for optimal solutions. The GA uses the solution set, not single solutions, in the solution space. Accordingly, the GA has the following advantages: the ability to increase the possibility of global optimization and the ability to define the performance index or evaluation function as intended by the designer.

To search for the optimal solution, multiple individuals will be generated; that is, solutions will be randomly selected from the solution set. This is called the initial population that is searched in order to find a solution through iterative selection, crossover, and mutation. To select excellent individuals, the evaluation function is used to evaluate how identical each individual is to the desired solution. Selection methods include the roulette wheel selection method, the expected-value selection method, the ranking selection method, and the tournament selection method.

In a study looking for an optimal solution through a GA, how to represent the optimal solution to a problem as a single individual and determining the standard for measuring how suitable each individual is to the desired optimal solution (i.e., the definition for the evaluation function) are the most important problems. And as the parametric values may affect the results, values will be predetermined for the GA. The procedure for the GA, which has been explained so far, is shown below [35].

Step 1 (generating the initial population). One of the potential solutions to a problem is randomly selected and becomes an individual, and several individuals combine to become the population.

Step 2 (evaluating and ending the test). Use the evaluation function to calculate the fitness value of one individual. If the number of iterations (n) reaches the maximum, end the test.

Step 3 (selection). Increase the probability of selecting individuals with a high fitness value and sample with replacement to reproduce the population.

Step 4 (genetic operators). Turn the selected individuals into a population with new information through crossover and mutation.

Step 5 (iteration). Replace the population having the new information with the initial population; then go back to Step 2.

2.3.2. The Genetic Algorithm Process. The GA has been frequently used to search for optimal paths, integrate data mining techniques, and determine optimal input variables. Among them, the studies on test data generation for software testing generate simulation data. In order to reduce the cost of manual software testing and concurrently increase the reliability of the testing processes researchers have tried to automate it [36].

The GA is more efficient than random testing in generating test data. Their efficiency will be measured as the number of tests required to obtain full branch coverage [19]. Srivastava and Kim [20] presented a method for optimizing software testing efficiency by identifying the most critical path clusters in a program using the GA to optimize the software path clusters. By identifying the most critical paths, testing efficiency can be increased. Singh [21] presented an algorithm for automatic generation of test data to satisfy path coverage and a basic process flow for generating test cases for path testing using the GA. His results show the efficiency of test data in terms of execution time and how it generates more effective test cases. Sharma et al. [22] presented a survey using a GA approach to address the various issues encountered during software testing and reported the results which showed that the performance of testing can be improved.

In the field of educational evaluation, some studies use the GA for test sheet composition. First, Hwang et al. [23] proposed two GAs to cope with the test sheet-generating problems. Experimental results show that test sheets with near-optimal discrimination degrees can be obtained in a much shorter time by employing GAs to the test sheet-generating problems. Lee et al. [37] applied the immune algorithm to test sheet problems in an attempt to improve the efficiency of composing near-optimal test sheets from item banks to meet multiple assessment criteria. The experimental results indicated that the immune algorithm is quite suitable for composing near-optimal test-sheets from large item banks. Li et al. [38] proposed a GA with an effective floating-point number coding strategy to generate a high quality test-sheet. In the experiment, the execution time, success ratio, and solution quality were compared to evaluate the performance of the proposed algorithm. Experimental results show that high quality test-sheets can be obtained in a much shorter time by employing a GA to test-sheet problems than other approaches. Xiong and Shi [25] presented the issue of test sheet generation and proposed a mathematical model and object function applying a GA to the issue of generating a test sheet for improving the quality of test sheets.

Second, Ou-Yang and Luo [26] improved traditional GAs and proposed a method of dynamically generating test sheets based on learners' learning situations. The experiments showed that speed and efficiency of composing test sheets that are more suitable for testing students' knowledge level were improved. Duan et al. [39] proposed an adaptive test sheet generation mechanism for determining candidate test items and conceptual granularities according to desired conceptual scopes, and an aggregate objective function applies the GA to find an approximate solution of mixed integer programming problem for test-sheet composition. Experimental results show that the adaptive test sheet generation mechanism can efficiently and precisely generate the various test sheets than the existing approaches in terms of various conceptual scopes, computation time, and item exposure rates. Previous studies found that the GA is effective in generating simulation data for software testing and composing test sheets.

3. Research Method

3.1. Dataset. This study evaluated the validity of the algorithms for generating item response data by comparing the item response data generated for simulation studies with real item response data. To this end, first, the difficulty and discrimination of each item based on the real item response data of students were calculated and sorted into criteria for comparison. The real data used for this study is the item response data consisting of 36 items. A total of 7,624 people participated in the study. We used four approaches to generating item response data to verify the effectiveness of the GA:

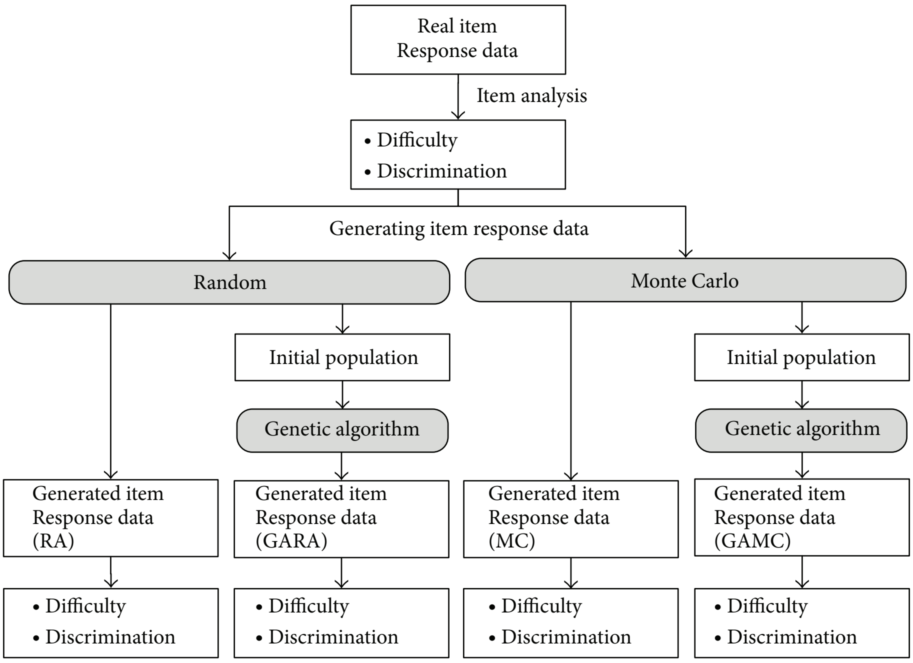


FIGURE 2: The four approaches to generating item response data.

(1) using only random method (hereinafter RA) to generate item response data;

(2) randomly generating initial population while using the GA (hereinafter GARA):

> randomly generating the initial population and using the GA based on real item parameters to generate item response data;

(3) using only Monte Carlo (hereinafter MC) to generate item response data:

> using MC based on actual item parameters to generate item response data;

(4) using the GA to generate the initial population with Monte Carlo (hereinafter GAMC):

> using Monte Carlo based on real item parameters to generate item response data and then using the GA to generate item response data.

The four approaches to generating item response data are shown in Figure 2.

To check if the generated item response data is similar to real item response data, root mean square error (hereinafter RMSE) was used as the measure for difficulty and discrimination based on classical test theory, and Kullback-Leibler divergence (hereinafter KLD) was used as the measure for difficulty and discrimination based on item response theory. The evaluation method is described in detail in Section 4.

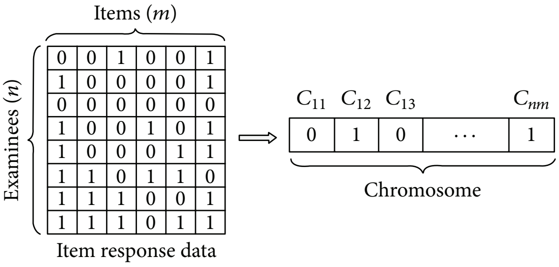


FIGURE 3: Item response data and chromosome structure design.

3.2. Chromosome Structure. The purpose of this paper is to generate item response data most similar to actual data. The first step of using the GA to find the optimal solution is to define the chromosome structure. As the optimal solution found by the GA is the item response data, it must be possible to express the item response data as a chromosome structure [35]. Item response data is generally configured as a two-dimensional array. As show in Figure 3, an item response data is defined as $n \times m$ matrix. A two-dimensional array as $n \times m$ matrix can be presented as nm-tuple for constructing a chromosome structure. It will be possible not only to use the item response data to obtain the final score of examinees, but also to get the parameters of each item (item difficulty, discrimination, and guessing). The array's columns represent the items of a test and the array's rows represent individual examinees. c_{ij} indicates the response of examinees i $(1, \ldots, n)$ on item j $(1, \ldots, m)$ as

$$c_{ij} = \begin{cases} 1: & \text{Examinee } i \text{ answers item } j \text{ correctly,} \\ 0: & \text{Other cases.} \end{cases} \quad (4)$$

The item response data is based on the responses made by n examinees to the questionnaire consisting of m items. In this paper, the item response data was converted into bit-strings and the chromosome structure was defined as shown in Figure 3. The item response data with $n \times m$ matrix, consisting of n examinees and m items, was converted into a chromosome made up of nm bits, and the nm-dimensional item response data was generated as a chromosome.

3.3. Fitness Function Design. The fitness function is a measure index applied to judge the quality of the generated item response data for the GA. Most studies, which apply GAs to an item response data, used item difficulty and item discrimination, that is, information on item characteristics, in applying the fitness function [11]. In order to compare an item response data based on the pretest results, the item difficulty and item discrimination of the real item parameters must be considered simultaneously to determine the fitness function. In this paper, the fitness function design is based on item difficulty and item discrimination:

$$\begin{aligned} \text{DiscErr} &= \sum_{i}^{n} \sqrt{\left(\widehat{a_i} - a_i\right)^2}, \\ \text{DiffErr} &= \sum_{i}^{n} \sqrt{\left(\widehat{b_i} - b_i\right)^2}, \end{aligned} \qquad (5)$$

where $\widehat{a}$, $\widehat{b}$ are the real discrimination and difficulty of the item, a and b are the discrimination and difficulty, respectively, derived from the generated item response data, i is the item number, and n is the total number of items.

This study defined the fitness function as the sum of the discrimination error and difficulty error as follows:

$$\text{Fitness} = \text{DiffErr} + \text{DiscErr}. \qquad (6)$$

3.4. Initial Population. In general, the size of the initial population varies depending on the complexity of the problem to solve. The initial population size in this study is set as 20 for the generation of a set of items response data. This study set the size of the initial population as 20. The two methods of setting initial chromosomes for experiments are (1) random and (2) Monte Carlo.

3.4.1. Randomly Generating the Initial Population. The algorithm for randomly generating the initial population accepts the number of items, the number of examinees, and the probability of answering an item correctly as input values. Random values are generated in a $U(0, 1)$ distribution. The number randomly generated in the uniform distribution is compared with the probability value entered as the probability of correctly answering the item. If the value of the number randomly generated in the uniform distribution is smaller than the resulting value of (2), it means that examinees correctly answered the item, and 1 is entered in u_{ij}. If the above number is greater than the resulting value, examinees do not know the correct answer of the item, so 0 is entered in u_{ij}. Randomly generating the initial population is shown in Algorithm 1.

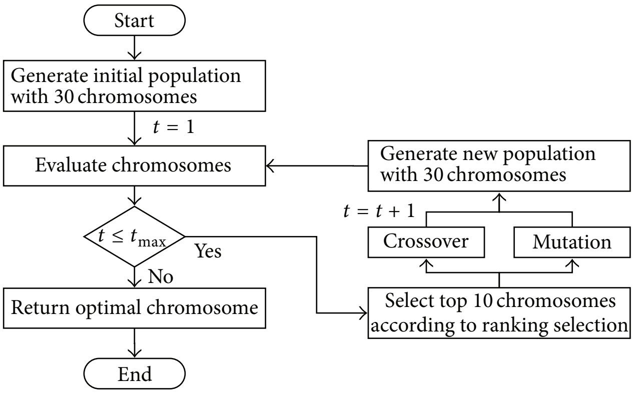

Start
Generate initial population with 30 chromosomes
t = 1
Evaluate chromosomes
t ≤ t_max
Yes
No
Return optimal chromosome
End
Generate new population with 30 chromosomes
t = t + 1
Crossover
Mutation
Select top 10 chromosomes according to ranking selection

Figure 4: Flowchart of a GA.

3.4.2. Using Monte Carlo to Generate the Initial Population Data. The Monte Carlo method accepts the ability of examinees (θ), item difficulty, and discrimination as input values. If item parameters and examinees' ability are set, the probability of the examinees' correctly answering each item will be determined. To specify whether examinees will answer each item correctly or incorrectly, generate a random number from the $U(0, 1)$ distribution. Compare the number randomly generated from the uniform distribution with the result of (2). If the value of the value randomly generated from the uniform distribution is smaller than the result of (2), it means that examinees answered the item correctly, so 1 is entered in u_{ij}. Otherwise, enter 0 in u_{ij}. In this way calculate the item response data for each examinee. See Algorithm 2.

3.5. Genetic Operation. The process of the GA is shown in Figure 4.

3.5.1. Selection. Selection operators model the phenomenon of natural selection; that is, well-adapted individuals survive and generate the next generation while ill-adapted individuals die out. Selections are made based on the fitness function, and though there are several selection methods, the basic principle is that individuals with a higher level of fitness will have more opportunities to be generated in the next generation. There are various selection methods that are widely used, such as the fitness proportionate selection method, the roulette wheel selection method, the expected-value selection method, the tournament selection method, and the elitist preservation selection [35]. This paper adopted the ranking selection method. The fitness of 30 individual chromosomes was calculated, with the top 10 chromosomes in terms of fitness left and carried over to the next generation.

3.5.2. Crossover. Crossover is used to generate new individuals by partially exchanging chromosomes between two individuals. Therefore, generation of individuals with better solutions through crossover is expected. In general, crossover is done when individuals exchange some genes. Depending on the types of coded genes, crossover can be defined differently. This study adopted the multipoint crossover method. Multipoint crossover involves two or more intersections.

```
Input: number of items ($n$), number of examinees ($m$), probability of being correct ($p$)
Output: item response data ($u$)
for all $i_m \in I$ do
    for all $s_n \in S$ do
        $g$ = Rand(0~1)    /* Generate a random real number between 0 and 1 */
        if $g < p$ then
            $u_{nm} \leftarrow 1$
        else
            $u_{nm} \leftarrow 0$
return $u$
```

Algorithm 1: Random generation (RG).

```
Input: number of items ($n$), number of examinees ($m$), item parameter vector (a: discrimination, b: difficulty), ability of
examinees ($\theta$)
Output: item response data ($u$)
for all $i_m \in I$ do
    for all $s_n \in S$ do
        $g$ = Rand(0~1)    /* Generate a random real number between 0 and 1 */
        $p \leftarrow 1/(1 + e^{-1.7*a*(\theta-b)})$
        if $g < p$ then
            $u_{nm} \leftarrow 1$
        else
            $u_{nm} \leftarrow 0$
return $u$
```

Algorithm 2: Monte Carlo generation (MG).

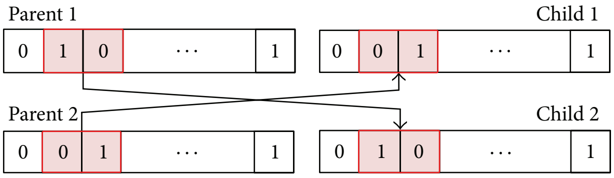


Figure 5: Crossover operation between two chromosomes.

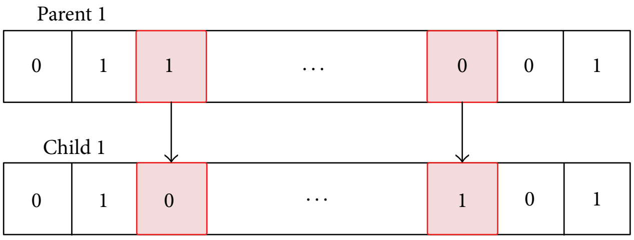


Figure 6: Mutation operation.

Two columns are randomly set up as intersections. Figure 5 illustrates a crossover where there are two intersections. In this study, 10 chromosomes, selected in the selection stage, were paired, crossover was done, and 10 new chromosomes were generated.

3.5.3. Mutation. As the population is generated repeatedly, the children will become similar to the population. As a result, even if crossover is conducted, new individuals may not be generated at times. Mutation makes up for the limitations of crossover. Mutation is used to apply a certain mutation probability to the genes of individuals and change the value of the alleles. It is a kind of local random search that generates new individuals. Specifically, mutation randomly picks a certain point in the chromosomes and changes its property. For example, if the selected value is 0, it will be changed to 1, and if it is 1, it will be changed to 0. This is illustrated in Figure 6; if two bits are chosen as mutation bit, the component values of each bit will be changed according to the probability of the mutation rate. For the purposes of this study, the mutation rate was set as 0.05.

4. Results and Discussion

The classical test theory is a method of using examinees' total scores to analyze items. The procedure is fairly straightforward and estimation can be calculated easily, but it has a weakness: depending on the characteristics of the group of examinees, the parameters of items (difficulty and discrimination) vary. According to item response theory, as the unique characteristics of an item are revealed because the estimated item parameters are unchanged due to the characteristics of the group of examinees, the precision of item analysis and estimation of examinees' ability will be enhanced. Accordingly, this study used not only the item parameters based on classical test theory, but also item parameters based on the item response theory to measure the accuracy of the algorithm.

According to classical test theory and item response theory, item analysis was conducted with regard to the item response data derived by the four methods (RA, GARA, MC, and GAMC), and the difficulty and discrimination of each item were obtained. We compared the real item parameters and the generated item parameters and carried out two experiments to evaluate the results of generated item response data by four approaches.

As a method of comparing item parameters, (1) RMSE was used for item parameters based on classical test theory, and (2) KLD was used for item parameters based on item response theory.

4.1. Measure Method

4.1.1. Root Mean Square Error. We adopted RMSE to compare item parameters based on classical test theory. RMSE is used when handling the difference between the estimated value (i.e., value predicted by a model) and the value observed in real-life situations. In RMSE, each difference is called a residual, and RMSE is used when residuals are synthesized with a single measure [40]. The RMSE of a prediction model with respect to the estimated variable X model is defined as the square root of the mean squared error:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} \left(X_{\mathrm{obs},i} - X_{\mathrm{model},i}\right)^2}{n}}, \tag{7}$$

where X_{obs} is the observed values and the X_{model} is the modeled values for item i.

4.1.2. Kullback-Leibler Divergence. We adopted the KLD as a measure for comparing item parameters based on item response theory. As with the method based on classical test theory, RMSE may also be used for comparing the difference between the real value and the estimated value of item parameters based on item response theory. However, unlike item parameters based on classical test theory, item parameters based on item response theory include the ability of the examinees. Accordingly, rather than simply comparing real values and observed values, as it is possible to consider the ability of examinees when the probability distribution obtained by parameters is compared, a more accurate comparison will be possible. KLD is used to calculate the difference between the two probability distributions [41]. Given the probability distributions P and Q for two random variables, the KLD of the two distributions is defined as follows:

$$D_{\mathrm{KL}} \left(P \parallel Q\right) = \sum_{i} P\left(i\right) \log \frac{P\left(i\right)}{Q\left(i\right)}, \tag{8}$$

where $P(i)$ is the random variable of the real item and $Q(i)$ is the random variable of the generated item.

RMSE compares the value difference between two finite groups composed of discrete components. In other words, the size of the value of RMSE is proportionate to not only the difference of each element, but also the number of elements. As the ICC based on the item response is a continuous probability variable, however, it has infinite elements, so we need to convert it into a discrete random variable for computer calculations. For a discrete random variable to be similar to a continuous random variable, we may increase the sampling size, but the computation cost will increase proportionately. For this study we calculated KLD for 400 sample points at an interval of 0.02 between −4 and 4 to handle the ICC as a discrete random variable. As RMSE and KLD are different from each other in terms of sample size, the values obtained using respective measurement methods are used to compare relative differences from the standard value within respective measurement methods.

4.2. Comparison of Item Parameters Based on Classical Test Theory. Table 1 illustrates the real item parameters of 36 items and the item parameter values obtained using the four methods (RA, GARA, MC, and GAMC). As shown in Table 1, the RMSE value of GAMC is 1.25, that of MC is 5.32, that of GARA is 14.56, and that of RA is 14.64. In the item response data, generated by Monte Carlo, difficulty was 1.07 and discrimination 4.25. The difference of discrimination was greater than that of difficulty. This seems to be because only the probability of examinees with a certain level of ability correctly answering the item was considered and also because the total scores of the examinees and the total score of other examinees were not considered when item response data was generated according to the Monte Carlo method. If the total score of examinees who correctly answered a certain item is high and that of examinees who incorrectly answered the item is low, the item has a high level of discrimination. Accordingly, if the total score of examinees is low, we should increase the probability of being correct for items with a low level of discrimination, and if discrimination is high, we should lower the probability of being correct so that it will be similar to real discrimination. As total scores are not considered in approaches using only Monte Carlo, however, there seems to be a large difference in discrimination.

When we used Monte Carlo to set the initial population and applied GAs, difficulty was lowered from 1.07 to 0.41, and discrimination was reduced from 4.25 to 0.84. It turned out that GAs made these values converge on the real item parameters. In other words, given an arbitrary item response data, it can be confirmed that real item response data will be found through crossover and mutation.

4.3. Comparison of Item Parameters Based on Item Response Theory. In item response theory, we can derive the item characteristic curve (ICC) by using the difficulty and discrimination values. The item characteristics curve is a probability distribution that indicates the probability of examinees' correctly answering items according to their abilities. The curve is determined by difficulty and discrimination. Figure 7 shows four ICCs. The horizontal axis is scaled in units of ability (θ) while the vertical axis is the probability of answering the item correctly. As show in Figure 7, GAMC is most similar to the item characteristic curve based on real

Table 1: Real classical test theory parameters and parameters obtained by RA, GARA, MC, and GAMC.

Item number	Real parameters		RA		GARA		MC		GAMC	
	β	α	β	α	β	α	β	α	β	α
1	0.50	0.37	0.49	0.21	0.49	0.21	0.51	0.49	0.50	0.39
2	0.38	0.37	0.50	0.22	0.50	0.21	0.32	0.46	0.36	0.39
3	0.57	0.46	0.50	0.22	0.50	0.22	0.59	0.61	0.56	0.45
4	0.51	0.36	0.49	0.18	0.49	0.21	0.52	0.48	0.51	0.29
5	0.45	0.33	0.50	0.23	0.50	0.22	0.43	0.44	0.45	0.34
6	0.74	0.46	0.50	0.19	0.50	0.20	0.80	0.56	0.77	0.48
7	0.72	0.38	0.49	0.20	0.49	0.21	0.80	0.48	0.75	0.39
8	0.62	0.57	0.51	0.19	0.51	0.19	0.64	0.69	0.61	0.58
9	0.42	0.35	0.50	0.23	0.50	0.23	0.39	0.48	0.42	0.37
10	0.75	0.43	0.50	0.21	0.50	0.23	0.81	0.51	0.75	0.40
11	0.67	0.53	0.49	0.26	0.49	0.26	0.70	0.65	0.66	0.53
12	0.66	0.49	0.50	0.19	0.50	0.18	0.70	0.61	0.67	0.50
13	0.66	0.49	0.49	0.21	0.49	0.21	0.70	0.62	0.65	0.49
14	0.59	0.51	0.49	0.22	0.49	0.22	0.61	0.65	0.59	0.53
15	0.62	0.44	0.50	0.23	0.50	0.22	0.66	0.58	0.62	0.44
16	0.68	0.52	0.49	0.23	0.49	0.24	0.73	0.65	0.69	0.52
17	0.75	0.59	0.50	0.19	0.50	0.18	0.78	0.69	0.73	0.59
18	0.63	0.51	0.50	0.21	0.50	0.21	0.66	0.64	0.63	0.52
19	0.43	0.42	0.50	0.23	0.50	0.23	0.41	0.53	0.43	0.44
20	0.79	0.50	0.49	0.23	0.49	0.23	0.83	0.58	0.81	0.51
21	0.46	0.47	0.49	0.23	0.49	0.23	0.44	0.58	0.44	0.47
22	0.41	0.34	0.49	0.20	0.49	0.20	0.38	0.45	0.41	0.35
23	0.58	0.52	0.50	0.22	0.50	0.21	0.60	0.66	0.58	0.53
24	0.73	0.56	0.50	0.21	0.50	0.20	0.77	0.67	0.72	0.52
25	0.76	0.59	0.49	0.21	0.49	0.22	0.79	0.67	0.61	0.30
26	0.52	0.41	0.50	0.18	0.50	0.18	0.52	0.55	0.52	0.37
27	0.76	0.61	0.50	0.23	0.50	0.27	0.78	0.71	0.74	0.55
28	0.48	0.32	0.49	0.21	0.49	0.21	0.47	0.44	0.47	0.34
29	0.57	0.56	0.49	0.21	0.49	0.23	0.59	0.70	0.57	0.57
30	0.73	0.59	0.49	0.21	0.49	0.20	0.76	0.70	0.72	0.59
31	0.67	0.60	0.50	0.24	0.50	0.24	0.69	0.73	0.67	0.61
32	0.77	0.56	0.49	0.19	0.49	0.20	0.81	0.66	0.77	0.55
33	0.58	0.57	0.49	0.23	0.49	0.23	0.60	0.70	0.58	0.58
34	0.66	0.50	0.50	0.21	0.50	0.21	0.69	0.64	0.66	0.51
35	0.58	0.54	0.50	0.20	0.50	0.20	0.60	0.68	0.58	0.56
36	0.52	0.56	0.50	0.23	0.50	0.23	0.53	0.69	0.53	0.57
RMSE			4.95	9.69	4.95	9.61	1.07	4.25	0.41	0.84
			14.64		14.56		5.32		1.25	

data. It is clear that the initial item response data, generated by MC, approaches the real item response data through GAs. The GARA shows the greatest difference, and the item response data approaches the real item response data in the initial population, but the speed of convergence is noticeably slow. The real item parameters based on the item response theory and item parameters based on the three methods are shown in Table 2.

Even if the GA is applied, the method of randomly setting the initial population was least accurate in both the classical test theory and item response theory. If the initial population is good in GAs, the probability of finding the optimal solution will increase [42, 43]. As shown in the table, if the initial population was generated randomly, difficulty was the value set as the probability for correctly answering all items. In this study, we randomly set it as 1/2, and, as a result, difficulty

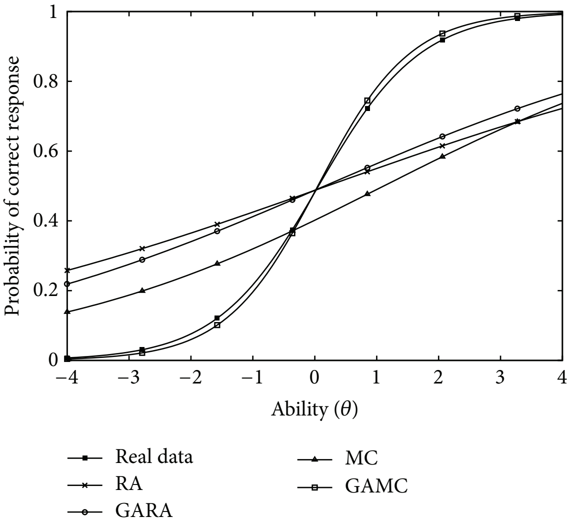


Figure 7: ICC graph for item number 1.

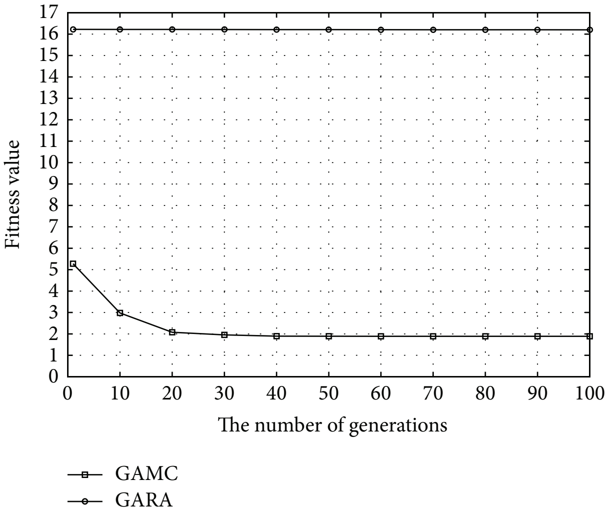


Figure 8: Fitness value change by generation.

converges to 0.5 with regard to all items. If the GA is applied to the initial population, fitness is reduced gradually, but the speed of convergence is slow.

4.4. Comparing the Speed of Convergence of the Fitness Functions in GAs. Table 3 presents the variation of fitness function values in accordance with generations increase in GARA and GAMC. Figure 8 illustrates the graph comparing the degree of fitness change by generation in GAs. GAMC converges to a certain value when the generation value reaches 50 if the fitness value rapidly decreases initially as generations increase. In contrast, in the GARA, fitness converges, but convergence speed is noticeably slow. The two methods apply the same algorithms, but the only difference lies in the method of setting the initial population. That is, the difference in the initial population affects the speed of finding optimal solutions in GAs.

As a matter of fact, the score distribution of examinees is a normal distribution, so there are many examinees around the mean, and the number of examinees diminishes as the total score increases or decreases. In the real item response data, there are examinees with diverse scores. However, if the population is randomly initialized, data will be uniformly distributed, and the uniform distribution will increase uncertainty. Saroj et al. [44] also mentioned that the poor initial population increases the running time for seeking optimal solutions. In other words, in this study, the initial population must be set up with various data so that the speed of finding optimal solutions can be improved. Accordingly, if item response data is generated using the GA, setting up the initial value according to the distribution of known item response

Table 2: Real item response theory parameters and parameters obtained by RA, GARA, MC, and GAMC.

Item number	Real parameters		RA		GARA		MC		GAMC	
	β	α	β	α	β	α	β	α	β	α
1	0.060	0.713	0.200	0.148	0.198	0.149	1.114	0.210	0.050	0.790
2	0.826	0.762	0.075	0.191	0.077	0.185	1.234	0.173	0.774	0.891
3	−0.247	1.022	0.097	0.186	0.097	0.196	1.635	0.183	−0.260	0.988
4	−0.011	0.675	0.232	0.130	0.120	0.172	1.057	0.241	−0.013	0.523
5	0.399	0.633	0.046	0.210	0.050	0.192	0.968	0.147	0.374	0.668
6	−1.110	1.143	0.047	0.145	0.048	0.142	1.733	0.181	−1.221	1.229
7	−1.251	0.842	0.228	0.160	0.165	0.179	1.310	0.174	−1.439	0.880
8	−0.392	1.507	−0.228	0.127	−0.211	0.137	2.209	0.161	−0.378	1.567
9	0.583	0.678	0.023	0.186	0.020	0.214	1.160	0.192	0.513	0.767
10	−1.204	1.046	0.019	0.165	−0.017	0.185	1.492	0.223	−1.363	0.935
11	−0.644	1.321	0.151	0.246	0.145	0.257	2.001	0.179	−0.650	1.314
12	−0.674	1.132	0.150	0.130	0.153	0.127	1.731	0.215	−0.721	1.192
13	−0.673	1.138	0.153	0.165	0.153	0.165	1.761	0.219	−0.662	1.151
14	−0.345	1.183	0.314	0.189	0.309	0.192	1.865	0.164	−0.348	1.282
15	−0.530	0.971	0.097	0.186	0.101	0.177	1.503	0.202	−0.564	0.950
16	−0.705	1.339	0.246	0.161	0.244	0.163	2.004	0.206	−0.764	1.281
17	−0.878	1.888	0.044	0.145	0.048	0.131	2.696	0.193	−0.861	1.717
18	−0.489	1.214	0.055	0.174	0.054	0.176	1.859	0.202	−0.497	1.277
19	0.471	0.851	0.102	0.192	0.101	0.194	1.371	0.203	0.390	1.015
20	−1.170	1.521	0.116	0.196	0.115	0.199	2.114	0.196	−1.315	1.495
21	0.287	1.037	0.185	0.192	0.176	0.195	1.646	0.212	0.304	1.126
22	0.661	0.668	0.185	0.139	0.183	0.141	1.067	0.138	0.607	0.708
23	−0.277	1.245	−0.071	0.209	−0.071	0.208	1.928	0.166	−0.284	1.284
24	−0.870	1.629	0.128	0.152	0.131	0.149	2.367	0.132	−0.914	1.336
25	−0.900	1.989	0.119	0.187	0.113	0.197	2.705	0.189	−0.836	0.528
26	−0.029	0.841	0.159	0.126	0.146	0.137	1.360	0.215	−0.068	0.738
27	−0.875	2.170	0.088	0.187	0.061	0.300	3.053	0.701	−0.930	1.505
28	0.237	0.585	0.257	0.158	0.260	0.156	0.940	0.191	0.259	0.659
29	−0.229	1.436	0.221	0.182	0.153	0.211	2.278	0.223	−0.223	1.469
30	−0.804	1.897	0.347	0.171	0.350	0.169	2.733	0.241	−0.785	1.679
31	−0.570	1.813	0.028	0.225	0.029	0.222	2.723	0.189	−0.574	1.707
32	−0.984	1.872	0.166	0.130	0.163	0.132	2.687	0.189	−1.076	1.567
33	−0.242	1.476	0.160	0.208	0.164	0.204	2.326	0.254	−0.240	1.541
34	−0.629	1.198	0.092	0.178	0.093	0.176	1.864	0.191	−0.662	1.210
35	−0.259	1.355	−0.030	0.160	−0.030	0.158	2.104	0.205	−0.251	1.407
36	−0.024	1.446	0.012	0.172	0.012	0.180	2.292	0.213	−0.045	1.520
	KLD		810.08		779.37		88.29		55.97	

data seems to be more effective than randomly determining the initial population.

5. Conclusion

The purpose of this study is to prove the effectiveness of the GA in generating item response data. To this end, we compared the item response data we generated using conventional Monte Carlo method with the item response data we generated using the GA. As comparison methods, we used RMSE for item parameters based on classical test theory and applied KLD to item parameters based on item response theory to compare the differences in the probability distribution.

The experiment results showed that the GA can be used to effectively create item parameters of generated item response data similar to real item parameters. Even though GAs are used, if the initial population is randomly set up, however, it was confirmed that the convergence speed is slow. As the random method does not guarantee the diversity of genes in two-dimensional item response data, the running cost for finding optimal solutions will increase. If the GA is applied for generating an item response data, it turned out to be most effective to set up the initial population with Monte Carlo

Table 3: Variation of fitness function value.

Generation	GARA	GAMC
1	16.2200	5.2779
10	16.2180	2.9801
20	16.2157	2.0759
30	16.2124	1.9571
40	16.2108	1.8969
50	16.2072	1.8909
60	16.2051	1.8890
70	16.2024	1.8890
80	16.2011	1.8890
90	16.2008	1.8890
100	16.1974	1.8890

and then apply the GA. In other words, the item response data, generated by the Monte Carlo, can be thought of as having gone through a process of seeking optimal solutions through the GA. This study found that we must use the GA to generate data similar to real item response data, but we must use Monte Carlo to generate the initial population. This study is meaningful in that we found that the GA contributes to generating more realistic data for simulation.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] M. I. Kellner, R. J. Madachy, and D. M. Raffo, "Software process simulation modeling: why? What? How?" *Journal of Systems and Software*, vol. 46, no. 2-3, pp. 91–105, 1999.

[2] A. Ammeri, W. Hachicha, H. Chabchoub, and F. Masmoudi, "A comprehensive literature review of mono-objective simulation optimization methods," *Advances in Production Engineering & Management*, vol. 6, no. 4, pp. 291–230, 2011.

[3] W. J. Stewart, *Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling*, Princeton University Press, Princeton, NJ, USA, 2009.

[4] D. S. Hira and P. K. Gupta, "Simulation and queuing theory," in *Operation Research*, S.Chand & Company, New Delhi, India, 2004.

[5] J. Banks and J. P. Spoerer, "Inventory policy for continuous review case: a simulation approach," *The Annals of the Society of Logistics Engineers*, vol. 1, pp. 51–61, 1986.

[6] M. Lu and S. M. AbouRizk, "Simplified CPM/PERT simulation model," *Journal of Construction Engineering and Management*, vol. 126, no. 3, pp. 219–226, 2000.

[7] R. Pisuchpen, "Integration of JIT flexible manufacturing, assembly and disassembly using a simulation approach," *Assembly Automation*, vol. 32, no. 1, pp. 51–61, 2012.

[8] Y. G. Sandanayake and C. F. Oduoza, "Dynamic simulation for performance optimization in just-in-time-enabled manufacturing processes," *International Journal of Advanced Manufacturing Technology*, vol. 42, no. 3-4, pp. 372–380, 2009.

[9] S. H. Cheraghi and M. Adashzadeh, "Comparative analysis of production control system through simulation," *Journal of Business and Economics Research*, vol. 6, no. 5, pp. 87–104, 2008.

[10] K. C. Seal, "Spreadsheet simulation of a queue with arrivals from a finite population: the machine repair problem," *International Journal of Operations and Production Management*, vol. 15, no. 6, pp. 84–100, 1995.

[11] Y. Luo, J. Li, Y. Xiao, and X. Lin, "A simulation study of overflow replacement policies for location management in mobile networks," *International Journal of Mobile Communications*, vol. 2, no. 2, pp. 103–127, 2004.

[12] E. A. Donkor and M. Duffey, "Optimal capital structure and financial risk of project finance investments: a simulation optimization model with chance constraints," *The Engineering Economist: A Journal Devoted to the Problems of Capital Investment*, vol. 58, no. 1, pp. 19–34, 2013.

[13] M. R. Harwell, "Analyzing the results of Monte Carlo studies in item response theory," *Educational and Psychological Measurement*, vol. 57, no. 2, pp. 266–279, 1997.

[14] M. Harwell, C. A. Stone, T.-C. Hsu, and L. Kirisci, "Monte Carlo studies in item response theory," *Applied Psychological Measurement*, vol. 20, no. 2, pp. 101–125, 1996.

[15] I. Spence, "Monte carlo simulation studies," *Applied Psychological Measurement*, vol. 7, no. 4, pp. 405–425, 1983.

[16] K. T. Han, "WinGen: Windows software that generates item response theory parameters and item responses," *Applied Psychological Measurement*, vol. 31, no. 5, pp. 457–459, 2007.

[17] T. I. Ören, "Impact of data on simulation: from early practices to federated and agent-directed simulations," in *Proceedings of the EUROSIM*, A. W. Heemink, L. Dekker, H. de Swaan Arons, I. Smit, and T. L. van Stijn, Eds., Delft, the Netherlands, June 2001.

[18] J. L. McCallum, D. Herckenrath, and C. T. Simmons, "Impact of data density and geostatistical simulation technique on the estimation of residence times in a synthetic two-dimensional aquifer," *Mathematical Geosciences*, vol. 46, no. 5, pp. 539–560, 2014.

[19] H.-H. Sthamer, B. F. Jones, and D. E. Eyres, "Generating test data for ADA generic procedures using genetic algorithms," in *Proceedings of the ACEDC '94*, pp. 134–140, PEDC, University of Plymouth, Plymouth, UK, 1995.

[20] P. R. Srivastava and T.-H. Kim, "Application of genetic algorithm in software testing," *International Journal of Software Engineering and Its Applications*, vol. 3, no. 4, pp. 87–96, 2009.

[21] B. K. Singh, "Automatic efficient test data generation based on genetic algorithm for path testing," *EARDA International Journal of Research in Engineering & Applied Sciences*, vol. 2, no. 2, pp. 1460–1472, 2012.

[22] C. Sharma, S. Sabharwal, and R. Sibal, "A survey on software testing techniques using genetic algorithm," *International Journal of Computer Science Issues*, vol. 10, no. 1, pp. 381–393, 2013.

[23] G.-J. Hwang, B. M. T. Lin, H.-H. Tseng, and T.-L. Lin, "On the development of a computer-assisted testing system with genetic test sheet-generating approach," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 35, no. 4, pp. 590–594, 2005.

[24] W. Fengrui, W. Wenhong, P. Quanke, and Z. Fengchao, "A novel online test-sheet composition approach using genetic algorithm," in *Proceedings of the 4th International Symposium on Advances in Computation and Intelligence, Huangshi, China, Ocotober 2009*, pp. 1–10, Springer, Berlin, Germany, 2009.

[25] L. Xiong and J. Shi, "Automatic generating test paper system based on genetic algorithm," in *Proceedings of the 2nd International Workshop on Education Technology and Computer Science (ETCS '10)*, pp. 272–275, March 2010.

[26] Y. Ou-Yang and H.-F. Luo, "Design of personalized test paper generating system of educational telenet based on genetic algorithm," in *Proceedings of the 4th International Conference on Computer Science and Education (ICCSE '09)*, pp. 170–173, Nanning, China, July 2009.

[27] K. T. Sun, Y. H. Lai, and B. C. Day, "A study on the test-sheet-generating strategy using genetic algorithm," in *Proceedings of the National Computer Symposium (NCS '99)*, pp. 379–386, 1999.

[28] A. Meng, L. Ye, D. Roy, and P. Padilla, "Genetic algorithm based multi-agent system applied to test generation," *Computers and Education*, vol. 49, no. 4, pp. 1205–1223, 2007.

[29] C. H. Papadimitrion and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall, Englewood Cliffs, NJ, USA, 1982.

[30] W. J. van der Linden, "Optimal assembly of psychological and educational tests," *Applied Psychological Measurement*, vol. 22, no. 3, pp. 195–211, 1998.

[31] F. M. Lord, *A Theory of Test Scores*, Psychometric Monograph no. 7, Psychometric Corporation, Richmond, Va, USA, 1952.

[32] R. M. Karp, "An introduction to randomized algorithms," *Discrete Applied Mathematics*, vol. 34, no. 1–3, pp. 165–201, 1991.

[33] M. Galdin and L. Laurencelle, "Assessing parameter invariance in item response theory's logistic two item parameter model: a Monte Carlo investigation," *Tutorials in Quantitative Methods for Psychology*, vol. 6, no. 2, pp. 39–51, 2010.

[34] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.

[35] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, Mass, USA, 1998.

[36] J. Edvardsson, "A survey on automatic test data generation," in *Proceedings of the 2nd Conference on Computer Science and Engineering in Linkoping*, pp. 21–28, ECSEL, October 1999.

[37] C.-L. Lee, C.-H. Huang, and C.-J. Lin, "Test-sheet composition using immune algorithm for E-learning application," in *New Trends in Applied Artificial Intelligence: Proceedings of the 20th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2007, Kyoto, Japan, June 26–29, 2007*, H. G. Okuno and M. Ali, Eds., vol. 4570 of *Lecture Notes in Computer Science*, pp. 823–833, Springer, Heidelberg, Germany, 2007.

[38] F. Li, F. Jing, X. Xie et al., "Estimation of multidimensional item response theory models in person parameter base on genetic algorithm," in *Proceedings of the International Conference on Anti-Counterfeiting, Security and Identification (ASID '10)*, pp. 207–210, Chengdu, China, July 2010.

[39] H. Duan, W. Zhao, G. Wang, and X. Feng, "Test-sheet composition using analytic hierarchy process and hybrid metaheuristic algorithm TS/BBO," *Mathematical Problems in Engineering*, vol. 2012, Article ID 712752, 22 pages, 2012.

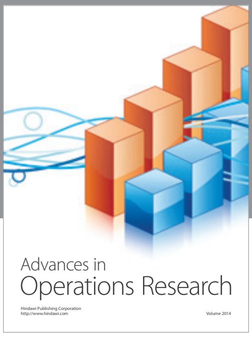
[40] C. J. Willmott and K. Matsuura, "On the use of dimensioned measures of error to evaluate the performance of spatial interpolators," *International Journal of Geographical Information Science*, vol. 20, no. 1, pp. 89–102, 2006.

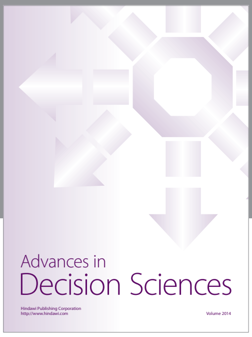
[41] S. Kullback, *Information Theory and Statistics*, Dover, New York, NY, USA, 1968.

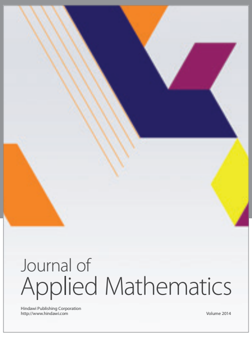
[42] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results," *Evolutionary computation*, vol. 8, no. 2, pp. 173–195, 2000.

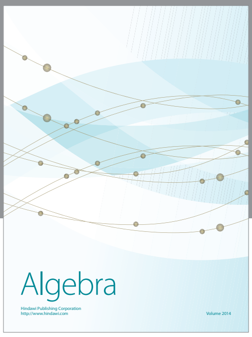
[43] E. K. Burke, S. Gustafson, and G. Kendall, "Diversity in genetic programming: an analysis of measures and correlation with fitness," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 1, pp. 47–62, 2004.

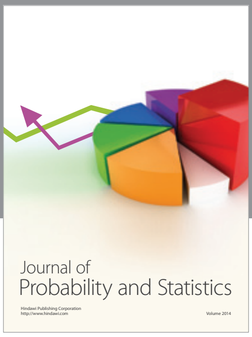
[44] Saroj, Kapila, D. Kumar, and Kanika, "A genetic algorithm with entropy based probabilistic initialization and memory for automated rule mining," in *Advances in Computer Science and Information Technology*, vol. 131 of *Communications in Computer and Information Science*, pp. 604–613, Springer, Berlin, Germany, 2011.


Advances in
Operations Research

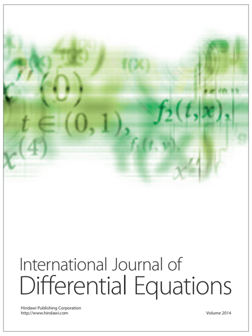

Advances in
Decision Sciences


Journal of
Applied Mathematics

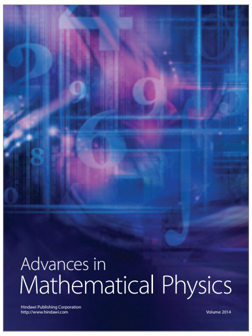

Algebra

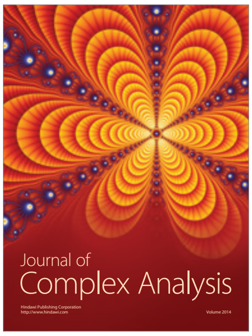

Journal of
Probability and Statistics

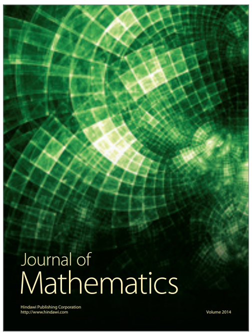
The Scientific
World Journal

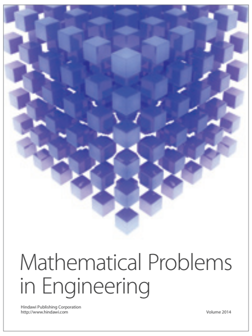

International Journal of
Differential Equations

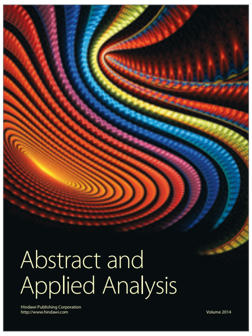
International Journal of
Combinatorics

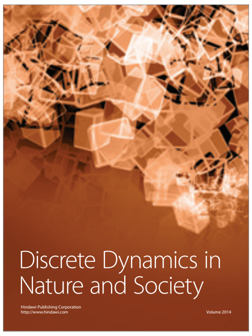

Advances in
Mathematical Physics

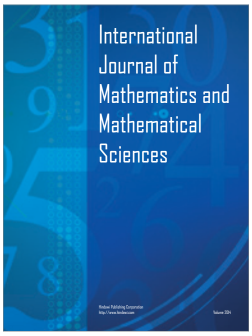

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

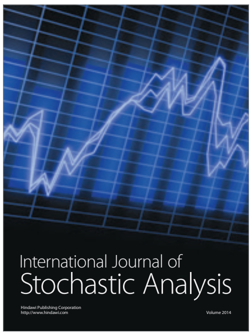

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization