An efficient genetic algorithm for item selection strategy

Koun-Tem Sun* Yi-Chun Lin Yueh-Min Huang
Dep. of Information and Learning Technology*
National University of Tainan*
Dep. of Engineering Science, National Cheng Kung University
ktsun@mail.nutn.edu.tw*

Abstract

Genetic algorithm has achieved remarkable results in solving the problem of combinatorial optimization in the artificial intelligence area in recent years. However, efficient search for one reasonable best solution is still underway among the massive restricted conditions. This paper proposes an efficient genetic algorithm based on Item Response Theory (IRT) and enables effective search for optimal solution or near optimal solution under restricted conditions. By modify the evolutionary parameters and the goal function of the simple genetic algorithm, test quality is not only acceptable by test designers, but the practicability is also enhanced. The proposed method is a more effective tool for education assessment researcher as it successfully extends artificial intelligence-genetic algorithm applied in the educational assessment.

1: Introduction

Item Response Theory (IRT) is a test theory based on degree of item difficulty, degree of distinctiveness, degree of speculation and test subject ability indicators used in conducting analysis. Under unidimensional assumption, the following advantageous are observed: test subjects' ability is assessed independently by items, degree of item difficulty and degree of discrimination is irrelevant to test subject sampling. The accuracy of ability assessment is assessable[10]. Also, in terms of test design, different test objectives are satisfactorily met. [1, 2, 7, 11, 20, 28] Therefore, when constructing the test, Item Response Theory not only surpasses Classical Test Theory, it is even likely to be replaced. During the process of test construction utilizing Item Response Theory, how test information function in selecting items and formulating a test that conforms to test formulators' objectives depend on item selection strategies used [18]. Currently, both international TOEFL Test and local basic academic tests contain question items from the existing question item bank and a set of test is then compiled. Therefore, more and more scholars are now devoted in studies related to item selection strategy problems [18, 12, 16, 17, 19].

An item selection strategy problem is a combinatorial optimization problem. Time required in calculation increases in multiple folds as the question item bank increases in number of items [14]. So far, scholars have proved it to be a NP-hard problem [24] and this type of question involves many calculation steps in formulating the best solution. Genetic algorithm may be the answer to it. This technique has been widely applied in scientific and engineering problems [13, 23, 3, 4, 5, 9, 21, 8]. What's more, some scholars have also applied this simple genetic algorithm on number of test items and test information function, as well as item selection strategy problems that do not have additional limitations with rather outstanding results. [18, [19] Nevertheless, since tests demand versatility and item selection often involves constraints under different objectives [5, 15, 25, 26, 27], this type of question is called constrained optimization problem [5]. When solving a constrained optimization problem, the penalty function approach in genetic algorithm is regarded as the most popular method. It is easy and executable; however, the major concern is that "can an appropriate penalty parameter be found?" [5, 6] Thus, scholar Kalyanmoy Deb has proposed another kind of evolutionary technique in solving this problem. Its features include: there is no need for penalty parameter. Tournament selection operator is used to pair up and make comparisons. Then, Euclidean Distance and mutational calculations are used to maintain reasonability and versatility of solutions. At the same time, better efficacy can be achieved [5]. Yet, no scholar has applied it in item selection strategy problems under constrained conditions.

In view of the above, simple genetic algorithm shows good results when used item selection strategy. However, there has not been an in-depth and clear discussion on the constrained conditions. Therefore, this paper has adopted the evolutionary technique in item selection strategy problems under multiple constrained conditions so as to ensure feasibility during item selection. In section 2, the simple genetic algorithm used in item selection will be introduced. In section 3, Deb genetic algorithm item selection strategy will be introduced. In section 4, hierarchical genetic algorithm item selection strategy will be introduced, in section 5, efficacy evaluation will be introduced. Finally, a conclusion is drawn with recommendations.

2: Simple genetic algorithm in item selection strategy

In applying the GA to item selection problem, each chromosome string represents a set of test k, which contain number n bits (referring to Number of test items in the item bank = n) among which, m is 1 (number of test items = m) and the rest are 0. x_i represents whether a test item is included in the test. ($x_i = 1$ Yes ; $x_i = 0$ No) For each chromosome, we may calculate the information capacity of target test and the information capacity deviation from the newly constructed test. The squared root sum of this deviation is defined as part of the fitness function during the evolutionary process.

$$E\left(X^k\right) = \sum_{j=1}^{s}\left(d_j - o_j^k\right)^2 \qquad (1)$$

Among which,

d_j refers to the target information value of ability level j

$j(=1 \sim s)$ refers to ability level; s refers to number of ability levels

$o_j^k = \sum_{i=1}^{n} w_{ij} x_i^k, \forall j = 1,\dots,s$

o_j^k refers to chromosomes k and its test information capacity in ability level j

$i(=1 \sim n)$ refers to item i in the question item bank ; n refers to number of question items in the item bank

w_{ij} refers to the information capacity of question item i in ability level j

$x_i^k \in \{0,1\}$ item selection i condition in k chromosome

Through genetic computations such as reproduction, crossover and mutation, more desirable offspring is produced. The evolutionary process is as shown in Fig.1.

(1) Setting the initial population of chromosome strings and parameters of the evolutionary process:

Randomly generate n number of binary string forming P number of chromosomes. Each chromosome of number is 1 and the rest are 0. The crossover, mutation, and reproduction probabilities are set to P_c, P_m, and P_r respectively. The evolutionary algebra is T , the initiation value is set at 0 and the maximum value is $gener\_no$.

(2) Calculating the value of fitness function for all the chromosomes strings k in the population and finding optimize fitness $chromosome_{opt}$:

$$\begin{aligned} fitness\left(X^k\right) &= E\left(X^k\right) + r^k \\ &= \sum_{j=1}^{s}\left(d_j - o_j^k\right)^2 + r^k \end{aligned} \qquad (2)$$

r^k number that satisfies $\sum_{i \in C_q} x_i > k_q$ or $\sum_{i \in C_q} x_i < k_q^{'}$

Among which,

$E\left(X^k\right)$ refers to target function

$q(=1 \sim p)$ number q constraint ; p refers to total number of constraints

C_q represents number q constraint including collective question items

k_q、$k_q^{'}$ setting value for number q constraint

r^k number of violating constraint

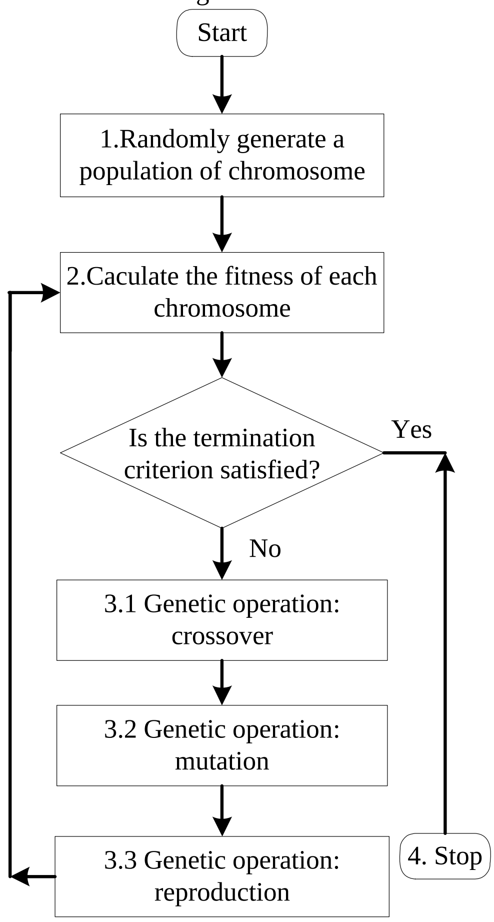

Figure1 . Flowchart for the GA evolutionary process

(3) For each chromosome k in the population, complete the following genetic operations for generating P offspring.

(3.1) Adopting two-point crossover, the offspring for each pair of parents is generated with the probability P_c. A section of the chromosome string in the offspring is the same as one parent and else is the same as the other parent. (As shown in Fig. 2)

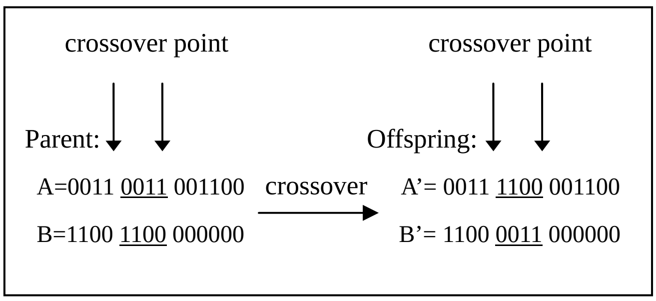

Figure2 . An illustration of the two-point crossover operation and 4 bits in the middle of chromosome strings which are exchanged in the offspring.

(3.2) Adopting two-point mutation (mutation probability: pm) randomly select single chromosomes from the parent group and choose two positions. Mutation occurs when 0 is changed to 1 and 1 is changed to 0. Gene then becomes the offspring. (As shown in Fig. 3)

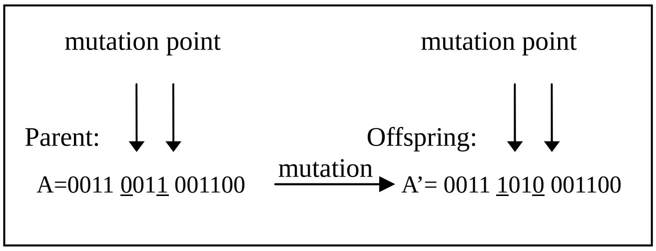

Figure3. An illustration of mutation operation for a chromosome with a 14-bit binary string, the two mutation points being randomly selected at the 5th and the 8th bit positions.

(3.3) The best chromosome strings are found in the "parent" population having the reproduction probability P_r. These then become the offspring which make up the new population.

(3.4) If the best chromosome in the offspring satisfies the requirements of the test designer, or the generation number reaches the maximum then the evolutionary process is stopped.

Else, the generation number is increased by one,

and the cycle continues again starting at step 2.

(4) Stop.

Finally, the optimize chromosome string is the optimal solution for item selection problems.

3: Deb Genetic algorithm in item selection strategy

Kalyanmoy Deb genetic algorithm is a penalty function approach that doesn't require penalty parameter; however, this evolutionary technique has not been applied in item selection strategy problems under restricted conditions. This method is mainly based on simple genetic algorithm combined with the evolutionary technique by Kalyanmoy Deb (Deb) to be used in item selection strategies under multiple restricted conditions. It differs from simple genetic algorithm in two aspects: one is fitness function calculation method and the other is crossover method in genetic calculation. In the crossover method, when the two are both reasonable solutions, further selection is done depending on condition setting. The distance between two entities and deviation scale after crossover serves as standards for such selection.

Advanced genetic algorithm parameter

By applying Debgenetic algorithm as item selection strategy, it will utilize genetic algorithm parameter in improving item selection quality. The parameter is described as follows:

(1) t value：prefix algebra, i.e. only prefix t is used for further selection determination

(2) Ed value：after crossover, the Euclidean Distance between two reasonable solutions ($P1$， $P2$)

$$Ed = \sqrt{\sum_{i=1}^{n}\left(x_i^{(P1)} - x_i^{(P2)}\right)^2} \qquad (3)$$

The major differences in item selection strategy when compared with genetic algorithm are described in the following:

(1) Fitness calculation：

$$fitness\left(X^k\right) = \begin{cases} E\left(X^k\right) & \text{,If: it is a reasonable solution} \\ f_{\max} + r^k & \text{,Or else:} \end{cases}$$

$$= \begin{cases} \sum_{j=1}^{s}\left(d_j - o_j^k\right)^2 & \text{, If: it is a reasonable solution} \\ f_{\max} + r^k & \text{, Or else} \end{cases}$$

r^k is the number of item that satisfies $\sum_{i \in C_q} x_i > k_q$ or $\sum_{i \in C_q} x_i < k_q^{'}$

$f_{\max}$ is the most inferior target function value of the reasonable solution; If no reasonable solution exists, then, $f_{\max} = 0$.

(2) Genetic calculation－crossover calculation:

Adopt two-point crossover method using tournament selection to select one from two entities. Then apply the following direction shown as follows:

(2.1) When two reasonable solutions are compared, selection is done further depending on the following rule.

$if\ (Ed < Ed_{\text{threshold}})\ then$ select smaller $E(X^k)$
$else$
the former combines with the reasonable solution to attain Ed
$if\ (Ed < Ed_{\text{threshold}})\ then$ the latter is selected
$else$ the former is selected

(2.2) When comparing a reasonable solution and an unreasonable solution, the reasonable solution is selected.

(2.3) When two unreasonable solutions are compared, the unreasonable solution that is violates the constraint to a lesser extent is selected.

4: Advanced Genetic algorithm in item selection strategy

Advanced genetic algorithm (AGA) item selection strategy is modified the parameters setting of Kalyanmoy Debgenetic algorithm parameter including parameters such as the distance between two new entity after crossover (Ed value), and prefix algebra（ t value. The main difference is Genetic calculation－crossover calculation：

Adopt two-point crossover method using tournament selection to select one from two entities. The comparison method is as follows:

(1) when two reasonable solutions are compared, selection is done further depending on the selection method used.

$if\ (t < t_{\text{threshold}})\ then$
$if\ (Ed < Ed_{\text{threshold}})\ then$ select smaller $E(X^k)$
$else$
the former combines with the reasonable solution to attain Ed

(2) when comparing a reasonable solution and an unreasonable solution, the reasonable solution is selected.

(3) When two unreasonable solutions are compared, the unreasonable solution with lesser violation of the constraint is selected.

5: Efficacy assessment

Since linear planning is most frequently applied in seeking optimized problems, this study has compared the efficacy of SGA, Deb, AGA, Greedy and Neural networks calculations with new linear planning (LP) [20] serves as reference for experimental results of LP. We coded a simulation tool to generate a virtual item bank containing 1000 three-parameter items and constraints. The values of the parameters and attributes in the constraints were randomly generated to approximate a uniform distribution among their ranges. In order to compare the performance of the proposed method, consistent conditions, the restricted conditions, basic settings of genetic algorithm parameter are identical with the LP experimental model as described in the following:

Question items in the question item bank	1000 items
Number of question items in the test	30 items
Evolutionary algebra	1000 algebra、1500 algebra
Target information capacity	100 set of single-peak and double-peak each for 5 different ability levels (as show in table 1 and table2)
Hierarchical genetic algorithm parameter	$Ed_{threshold} = 6$、$t_{threshold} = 75$
chromosome	100 sets
crossover rate	p_c =80%
mutation rate	p_m =10%
reproduction rate	p_r =10%
Restricted condition	more then 10 content attributes

This research enhances the mutation rate to 10% in order to enables each offspring have enough variability and avoid premature convergence, moreover, have the more rapidly converging speed and solution quality in 1000 evolutionary generations.

If the item bank contains 1000 questions, and 40 items are to be selected, the restricted conditions are: 10 + 5 + 6 + 1 + 1000 =1022. So, the selected items must satisfy these 1022 constraints. Using various genetic algorithms, testing has been conducted 10 times to attain the mean of the squared root sum deviation.

Experimental results：

1. The deviated values of SGA, Deb, AGA, Greedy and Neural networks are all lesser than LP
2. AGA deviated value improvement rate is good. As compared to LP, the mean improvement rate is as high as 99.96%.(As shown in table3)

6: Conclusion and recommendation

In the paper, we have improved the evolutionary technique of simple genetic algorithm, parameter setting strategy and conducted “multiple restricted conditions” item selection strategy. Also, the efficacy and feasibility have been assessed. In accordance with study results, the following concluding statements have been drawn:

(1) The algorithm used in this paper effectively solved the problem of “multiple restricted condition” item strategy. The minimum deviated value is attained to provide a more practical technique that ensures better test quality that is acceptable by the test designer requirements.

(2) In generally, simple genetic algorithm only needs the fitness function information, moreover does not need other auxiliary information. Therefore, it should uses the various state fitness function, and save the computer resources to avoid the complicated mathematics operation [30]. Based on simple genetic algorithm, Kalyanmoy Deb evolutionary technique has been successfully combined. The “parameter-setting strategy” has also been improved. “Ed value” has been used to add versatility and increase the chance of convergence of “t parameter ” (to close the optimal value) During the evolutionary process, item selection problems have been successfully solved which greatly reduced the deviated value between constructed test information function and target information function. As compared to another new study-LP, results show that the advanced genetic algorithm proposed in this study produced better results than conventional LP. The improvement rate reaches as high as 99.96%. As compared to the method proposed by Kalyanmoy Deb, the improvement rate also reaches as high as 62.12%. Genetic algorithm is not only used extensively under multiple restricted conditions, it also provides education assessors a more effective

tool and a practical technique to use.

References

[1]Ackerman, T. (1989). *An alternative methodology for creating parallel test forms using the IRT information function.* The Annual Meeting of the National Council for Measurement in Education, San Francisco.

[2]Baker, F. B., Cohen, A. S., & Barmish, B. R. (1988). *Item characteristics of tests constructed by linear programming.* Applied Psychological Measurement, 12(2), 189-199.

[3]Bramlette, M. F. (1991). *Initialization, mutation and selection methods in genetic algorithms for function optimization.* In R. K. Belewand L. B. Booker, eds., Proceedings of the Fourth International Conference on Genetic Algorithm, Morgan Kaufmann.

[4]Toroslu, H. T. & Arslanoglu, Y. (2006). *Genetic algorithm for the personnel assignment problem with multiple objectives* . Information Sciences, (in press).

[5]Deb, K. (2000). *An efficient constraint handling method for genetic algorithms.* Computer Methods in Applied Mechanics and Engineering, 186(2-4), 311-338.

[6]Deb. K. (2001). *Genetic Algorithms for Optimization.* (Rep. No. 2001002). India: Department of Mechanical Engineering Indian Institute of Technology Kanpur, Kanpur Genetic Algorithms Laboratory(KanGAL).

[7]De Gruijter, D. N. M. (1990). *Test construction by means of linear programming.* Applied Psychological Measurement, 14, 175-181.

[8]Deb, K., Pratap, A., Agrawal, S., & Meyarivan, T. (2002). *A fast and elitist multiobjective genetic algorithm: NSGA–II.* IEEE Transactions on Evolutionary Computation, 6(2), 182–197.

[9] G. Levitin, (2005). *Weighted voting systems: reliability versus rapidity*. Reliability Engineering & System Safety, 89(2) pp. 177-184.

[10]Hambleton, R. K., & Swaminathan, H. (1985). *Item Response Theory: Principles and Applications.* Hingham, MA: Kluwer, Nijhoff.

[11]Lord, F. M. (1977). *Practical applications of item characteristic curve theory.* Journal of Educational Measurement, 14, 117-138.

[12]Luecht, R. M., & Hirsch, T. M. (1992). *Item selection using an average growth approximation of target information functions.* Applied Psychological Measurement, 16(1), 41-51.

[13]Metcalfe, T. S., & Charbonneau, P. (2003). *Stellar structure modeling using a parallel genetic algorithm for objective global optimization.* Journal of Computational Physics, 185, 176.

[14]Papadimitrion, C. H., & Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity.* Prentice-Hall, Inc., NJ:Englewood Cliffs.

[15]Sanders, P. F., & Verschoor, A. J. (1998). *Parallel Test Construction Using Classical Item Parameters.* Applied Psychological Measurement, 22(3), 212-223.

[16]Stocking, M. L., & Swanson, L. (1993a). *A method for severely constrained item selection in adaptive testing.* Applied Psychological Measurement, 17(3), 277-292.

[17]Stocking, M. L., & Swanson, L. (1993b). *A model and heuristic for solving very large item selection problems,* Applied Psychological Measurement, 17(2), 151-166.

[18]Sun, K. T. (1999). *An Effective Item Selection Method by Using AI Approaches. Advances in Intelligent Computing and Multimedia Systems,* 137-142, Baden-Baden, Germany.

[19]Sun, K. T. (2000a). *A Genetic Approach to Parallel Test Construction.* International Conference on Computers in Education 2000, pp.83-90, The Grand Hotel, Taipei, Taiwan.

[20]Sun, K. T. (2002). *A Genetic Algorithm for Parallel Test Forms (Rep. No. 00101).* Taiwan, Tainan: National Tainan Teachers College, AI & ICAI Lab.

[21] Salcedo-Sanz S, Bousono-Calzon C, Figueiras-Vidal AR (2003) *A mixed neural-genetic algorithm for the broadcast scheduling problem.* IEEE Trans Wirel Commun, 2(2), 277-283.

[22]Theunissen, T. J. J. M. (1985). *Binary programming and test design.* Psychometrika, 50, 411-420.

[23]Tsai , H. K. , Yang , J. M., Tsai, Y. F., and Kao, C. Y. (2004). *A genetic algorithm for large traveling salesman problems.* IEEE Transactions on Systems, Man and Cybernetics - Part B , 34 , 1718-1729.

[24]van der Linden, W. J. (1998). *Optimal assembly of psychological and educational tests.* Applied Psychological Measurement, 22(3), 195-209.

[25]van der Linden, W. J., & Boekkooi-Timminga, E. (1989). *A maximum model for test design with practical constraints.* Psychometrika, 54, 237-247.

[26]van der Linden, W. J., & Reese, L. M. (1998). *A model for optimal constrained adaptive testing.* Applied Psychological Measurement, 22(3), 259-270.

[27]Wightman, L. F. (1998). *Practical issues in computerized test assembly.* Applied Psychological Measurement, 22(3), 292-302.

[28]Wright, B. D., & Douglas, G. A. (1977). *Best procedures for sample-free item analysis.* Applied Psychological Measurement, 1, 281-295.

[29]Wright, B. D., & Stone, M. H. (1979). *Best test design.* Chicago: MESA.

[30]Levitin G.. (2006). *Genetic algorithms in reliability engineering*, Reliability Engineering & System Safety, 91(9), 975-976.

	Index of Ability Level				
	1	2	3	4	5
Ability Level	-2.0	-1.0	0.0	1.0	2.0
Target Test Information	4	6	12	6	4

Table1. An example of the target test information function (one-peak shape) with ability levels ranging from –2 to 2.

	Index of Ability Level				
	1	2	3	4	5
Ability Level	-2.0	-1.0	0.0	1.0	2.0
Target Test Information	4	10	4	10	4

Table2. An example of the target test information function (two-peak shape) with ability levels ranging from –2 to 2.

Item \ Algorithm	AGA	LP	SGA	Deb	Greedy Approach	Neural Network
one-peak shape distribution	0.000883	1.508440	0.021207	0.002153	0.6986	0.7416
two-peak shape distribution	0.000873	2.551810	0.030765	0.003151	0.7251	0.6945
Average of squared error	0.000878	2.030125	0.025986	0.002652	0.7119	0.7181
Improvement（%）	**---**	**99.956754**	**96.621392**	**66.894796**	**98.2664**	**98.6343**

Improvement（%）： $\left(error_x - error_{AGA}\right) / error_x \times 100$

$error_{AGA}$ ：the errors(deviations) generated by AGA approach.

$error_x$ ：the errors(deviations) generated by LP, SGA , Deb , Greedy or Neural Network approach.

Table3. The average deviation and improvement ratio for each solution generated by different methods. (1000-item)