A Novel Genetic Algorithm for Constructing Uniform Test Forms of Cognitive Diagnostic Models

Ye-shi Jiang[2,3] Ying Lin[1,2] Jing-Jing Li[4] Zheng-jia Dai[1] Jun Zhang[2,5,6] Xinglin Zhang[5]

[1] Department of Psychology, Sun Yat-sen University
[2] School of Data & Computer Science, Sun Yat-sen University
[3] Key Laboratory of Machine Intelligence & Sensor Networks, Ministry of Education, P.R. China
[4] School of Computer Science, South China Normal University
[5]School of Computer Science & Engineering, South China University of Technology
[6]State Key Laboratory of Mathematical Engineering and Advanced Computing

Abstract – **Cognitive diagnostic models (CDMs) are a new class of test models developed for educational assessment. They have gained growing attention in recent years for their distinctive ability to provide detailed feedback about examinees' ability. Automatic test assembly (ATA), as in other test models, has been one of the most critical issues in the development and applications of CDMs. However, developing ATA methods for CDMs is especially challenging because no close-form expressions can measure the quality of a test form based on the items used. Although some heuristic methods have been proposed for building a single test form of CDMs, few ATA methods can construct uniform test forms of CDMs, in which each test form contains a different set of items but meets equivalent demand of test quality. In order to fill the gap, this paper proposes a novel genetic algorithm (GA) for constructing uniform test forms of CDMs. The effectiveness and efficiency of the proposed method is validated on a synthetic item pool under different conditions.**

Index Terms – automatic test assembly (ATA); uniform test forms; cognitive diagnosis model (CDM); genetic algorithm (GA)

I. INTRODUCTION

Educational assessment usually refers to use test forms to evaluate, measure, and document examinees' knowledge, skills, attributes, and ability. In recent years, automatic test assembly (ATA), that is, building test forms without human intervention, has been a popular research topic in educational measurement. In particular, due to the increasing demand of test security and reliability, how to generate uniform test forms that use different sets of items to achieve similar test quality has begun to draw growing attention.

In the context of classic test theory (CCT) and item response theory (IRT), a number of different methods have been proposed to generate uniform test forms. In 1989, literature [1] proposed a basic sequential method that builds one test form at a time until the desired number of test forms is met. The items used in previous tests are excluded from the candidate item set for the next test form. The chief disadvantage of this sequential method lies in unbalanced test quality as test forms generated later may be less satisfying due to the lack of candidate items. Such a disadvantage becomes more obvious when the desired number of test forms increases. In order to address the disadvantage, [2] and [3] proposed parallel test construction methods that generates multiple test forms simultaneously using liner programming (LP) and network-flow programming (NFP), respectively. Reference [4] proposed an improved sequential method, namely "the cell-only method", which first clusters the given items into different groups based on similarity of item characteristics, and then selects items from each group at random to generate different test forms of similar quality. As a further improvement, [5] incorporated a flexible content-balancing element into the cell-only method. Considering the fact that the generation of uniform test forms can be converted to a combinatorial optimization problem, metaheuristics are also applied, including the variable neighborhood search, tabu search, and evolutionary computation methods such as genetic algorithm (GA), bee algorithm, and adapted clonal selection algorithm [6]-[13], etc.

However, all the above methods are only applicable for generating test forms of CTT and IRT, which describe examinees' overall performance with a score on a continuous scale. For targeted and individualized teaching and learning, both teachers and students desire more detailed feedback, instead of a single score, to be obtained from a test. Encouraged by the desire, [14] proposed a new class of test models, namely cognitive diagnostic models (CDMs). Unlike CCT and IRT, CDMs are latent class models that formulate the probability of correctly answering an item as a function of an attribute mastery pattern. Hence, given an examinee's response to a test form, the examinee's attribute mastery pattern can be estimated. Due to the essential difference between CDMs and traditional test models, the ATA methods for CCT and IRT are no longer applicable for CDMs. Reference [15] proposed a discrimination index, named cognitive diagnostic index (CDI), to measure the ability of an item to distinguish different attribute mastery patterns. Based on CDI, a ATA method for CDMs was proposed, which selects items with the largest CDI to compose a test form with high overall diagnostic accuracy. Reference [16] introduced a GA to build a test form that can satisfy different diagnostic purposes. Reference [17] modelled the test construction problem using binary programming (BP) and obtained better diagnostic accuracy at the attribute level.

Despite the above progress in ATA for CDMs, no ATA methods can construct uniform test forms of CDMs effectively and efficiently. In order to fill the gap, this paper proposed a GA-based ATA method for solving the problem. Given a desired number of test forms and an item pool, the proposed

This work was supported in part by NSF of China Project No. 61309003, No. 61332002, No. 61300044, and the open project of the State Key Laboratory of Mathematical Engineering and Advanced Computing. Ying Lin (yinglinchn@gmail.com) and Jun Zhang (junzhang@ieee.org) are the corresponding authors.

method encodes a set of test forms as a chromosome. The chromosome is evaluated from two aspects: the average and the difference of the quality of test forms. Note that since no close-form expressions exist to measure the quality of a test form based on the item used, the test quality is evaluated on the basis of simulation that involves virtual examinees. The proposed method then evolves a population of different chromosomes towards the optimal set of test forms through evolutionary operators specially designed for the problem. In order to show the effectiveness and efficiency of the proposed method, its performance is validated on a synthetic item pool under different conditions. The results are compared with a baseline method based on CDI.

The remainder of this paper is organized as follows. In section II, a brief introduction of CDMs is presented. In section III, the proposed GA-based ATA method is described in detail. In section IV, the proposed method is validated on a synthetic item pool and the results are compared to a baseline method based on CDI. In the end, conclusions are drawn and further research directions are suggested in Section V.

II. Background

CDMs are a class of statistical models that formulate the probability of answering an item correctly given an attribute master pattern. Using CDMs, the attribute mastery pattern of an examinee can be estimated, which classifies the examinee into a number of discrete mastery levels, usually two (master or not), on each of the K attributes in consideration. By doing so, CDMs can measure the examinee's mastery status thoroughly and thus reveal his/her strengths and weakness at the attribute level [16][18] .

In CDMs, the attribute mastery pattern of an examinee j is represented by a vector $\boldsymbol{\alpha} = (\alpha_1,\alpha_2,...,\alpha_K)$, where each variable α_k is the mastery level of the kth attribute, $k = 1,2,...,K$, and K is the number of attributes in consideration. Specifically, in the case of two mastery levels, $\alpha_k = 1$ indicates mastery of attribute k while $\alpha_k = 0$ indicates the opposite. In total, there are 2^K possible attribute mastery patterns.

A fundamental component of CDMs is the Q-matrix. The Q-matrix is an $I \times K$ matrix, where each element $q_{ik} \in \{0,1\}$ indicates whether attribute k is measured by item i ($q_{ik} = 1$) or not ($q_{ik} = 0$), $i = 1,2,...,I$, and I is the number of items in use. Once the Q-matrix has been specified and items have been administered in a field test, each examinee with a specified mastery pattern will form a response pattern $\boldsymbol{x} = (x_1,x_2,...,x_I)$, where $x_i = 1$ indicates the examinee correctly answers item i and $x_i = 0$ for the opposite.

In order to estimate the attribute mastery pattern of an examinee based on his/her response $\boldsymbol{x}$ to a test $\boldsymbol{u}$, [16] proposed the following estimation scheme. First, compute the posterior probability of obtaining the response $\boldsymbol{x}$ given that the attribute master pattern of the examinee is $\boldsymbol{\alpha}$:

$$P(\boldsymbol{x}|\boldsymbol{\alpha}) = \prod_{i=1}^{I} P(x_i = 1|\boldsymbol{\alpha})^{x_i} \left[1 - P(x_i = 1|\boldsymbol{\alpha})\right]^{1-x_i} \quad (1)$$

After calculating the posterior probability of $\boldsymbol{x}$ for each possible attribute master pattern, the examinee's estimated attribute master pattern $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_1,\hat{\alpha}_2,...,\hat{\alpha}_K)$ can be obtained by minimizing the posterior expected error rate as

$$\hat{\boldsymbol{\alpha}} = \min^{-1}_{\boldsymbol{\alpha}} \sum \left[P(\boldsymbol{x}|\boldsymbol{\alpha}) \left(\sum_{k=1}^{K} |\alpha_k - \hat{\alpha}_k| \right) \right]. \quad (2)$$

Many candidate CDMs have been proposed in not only the field of education assessment, but also the field of psychometric [19], including the restricted latent class model which is later called the DINA model (deterministic input, noisy "and" gate model), generalization of the DINA model (G-DINA), noisy input, deterministic "and" gate (NIDA) model, compensatory multiple classification latent class model (MCLCM), and reparametrized unified model (RUM). In this paper, the reduced version of RUM is used in simulation.

The reduced RUM [20][21] assumes that all the items are dichotomously scored as correct or incorrect. Let π_i^* denote the probability of correctly applying all the attributes that are required for item i by an examinee who has mastered all of these attributes. To quantify the decrement in the probability due to non-mastery of certain attributes, let π_{ik} denote the probability of correctly responding to item i for examinees who has mastered attribute k and r_{jk} denote the corresponding probability for a non-master of attribute k. Note that $\pi_{ik} \geq r_{jk}$ because masters are assumed to have a greater chance to get a correct answer than non-masters do, and $r_{jk}^* = r_{jk} / \pi_{ik}$ is the ratio of r_{jk} to π_{ik}. A smaller r_{jk}^* value implies that item i has a higher distinguish ability between masters and non-masters on attribute k. Given π_i^* and r_{jk}^* ($k = 1,2,...,K$), the probability for an examinee with attribute master pattern $\boldsymbol{\alpha}$ to correctly answer item i is:

$$P(x_i = 1|\boldsymbol{\alpha}) = \pi_i^* \prod_{k=1}^{K} r_{ik}^{*(1-\alpha_k)q_{ik}} \quad (3)$$

From the above (3), it can be seen that if $\alpha_k = 1$ for all k being measured by item i, the probability of answering item i correctly equals π_i^*; otherwise, non-mastery of attribute k reduces the probability by a factor r_{jk}^*. Compared to the typical RUM, the reduced RUM omits the examinee's "supplemental ability" that may affect performance on the item but not exists in the Q-matrix. The reduced RUM is also used for simulation in [16], [17], and [22].

III. The Proposed Method

In this section, the proposed GA-based ATA method is introduced. The designs of the fundamental components of GA, including the representation scheme, fitness function, and genetic operators are described in detail.

A. Representation Scheme

The genetic algorithm (GA), originally proposed by [23], is a global search metaheuristic that simulates the evolutionary mechanism in nature. In order to represent a solution to the problem of generating uniform test forms of CDMs, each chromosome in the population of GA is defined as an M by I matrix $\boldsymbol{M}_t$, where M is the required number of test forms, I is the number of candidate items, the (u,v) entry $m_{uv}^{(t)}$ in $\boldsymbol{M}_t$ is an indicator of whether the ith item is selected into the uth test form ($m_{ui}^{(t)}=1$) or not ($m_{ui}^{(t)}=0$), t = 1,2,...,NP, and NP is the population size. Let $\boldsymbol{P}(g)$ = {$\boldsymbol{M}_1$, $\boldsymbol{M}_2$, …, $\boldsymbol{M}_{NP}$} denote the population of the gth generation, g = 0,1,.... The initial population $\boldsymbol{P}(0)$ is generated by randomly assigning values (1 or 0) to the genes for each chromosome. Note that since the number of items selected into each test form (indicated as $l_u=\sum_{i=1}^{I} m_{ui}$) must meet the test length constraint (i.e. l_u = *Length*), a random modified method is applied to randomly delete extra items from the selected items if l_u > *Length*. Otherwise randomly select more items from the item pool if l_u< *Length*. Then, a chromosome that fits the test length constraint but items are overused (i.e., the number of test forms that use the item exceeds the given overlapping threshold), the repair strategy introduced in Part C is used to address the overlapping constraint.

B. Fitness Function

A fitness function is required in GA to evaluate how good the solution is. In the problem of generating uniform test forms of CDMs, a solution is evaluated from two aspects: 1) the overall quality of test forms and 2) the variation between the quality of test forms. In order to realize the above evaluation, a measure to the quality of a test form is needed at first. In this paper, the three test quality evaluation criteria proposed by [16] are adopted. Detailed definitions of the three criteria are given below.

In CDMs, each examinee is categorized into one of the two levels (mastery and non-mastery) on each of the K attributes. The classification error rates are thus a fundamental way to measure test quality. For an examinee with a true attribute mastery pattern $\boldsymbol{\alpha}$ = (α_1,α_2,...,α_K), the number of classification errors is $\sum_{k=1}^{K}\left|\alpha_k-\hat{\alpha}_k\right|$, where $\hat{\alpha}_k$ denotes attribute k of the estimate attribute pattern $\hat{\boldsymbol{\alpha}}$. The expected error rate for attribute k can thus be obtained as follows:

$$e_k=\sum_{\boldsymbol{\alpha}} P(\boldsymbol{\alpha})E_{\boldsymbol{\alpha}}(\left|\alpha_k-\hat{\alpha}_k\right|), \tag{4}$$

where $P(\boldsymbol{\alpha})$ is the prior probability for an examinee to have the attribute mastery pattern and $E_{\boldsymbol{\alpha}}()$ denotes the expected value under $\boldsymbol{\alpha}$. Based on the expected posterior error rates e_1,e_2,...,e_K, the three evaluation criteria for test quality are defined as follows:

$$F^{(1)}=\sum_{k=1}^{K} e_k \tag{5}$$

$$F^{(2)}=\max_{k=1}^{K} e_k \tag{6}$$

$$F^{(3)}=\sum_{k=1}^{K}\left|e_k-\varepsilon_k\right| \tag{7}$$

where ε_k is the targeted error rate on attribute k.

As can be seen from Eq. (5) to Eq. (7), the three criteria are respectively for the diagnosis purpose of minimizing (a) the overall classification error rate, (b) the maximum classification error rate across all the attributes, and (c) the sum of the absolute difference between actual and targeted error rates. For the last purpose, the set of target error rates ε_1,ε_2,...,ε_K is determined in prior.

Based on the above evaluation criteria for test quality, the overall quality of the set of test forms represented by a chromosome $\boldsymbol{M}_t$ can be measured as

$$f_{\text{qua}}=\max_{1\le u\le m} F_u^{(\,)}, \tag{8}$$

where $F_u^{(\,)}$ is the quality of test form u in $\boldsymbol{M}_t$ according to one of the three evaluation criteria. The variation among the test forms is calculated as

$$f_{\text{var}}=\frac{1}{M}\sum_{u=1}^{M}[F_u^{(\,)}-\bar{F}^{(\,)}]^2, \tag{9}$$

where $\bar{F}^{(\,)}$ is the average quality of all the M test forms on the corresponding evaluation criterion.

The goal of the proposed GA-based ATA method is to improve the quality of uniform test forms (i.e. minimizing f_{qua}) while reducing the quality variation among the test forms (i.e. minimizing f_{var}). In order to eliminate the difference between the scales of f_{qua} and f_{var}, the fitness function is defined as

$$\max f=exp\left(-f_{\text{var}}\right)+exp\left(-f_{\text{qua}}\right). \tag{10}$$

By doing so, a larger fitness value indicates a better solution to the problem in consideration.

C. Genetic Operators

The following three genetic operators are used in the proposed GA-based ATA method.

1) Selection: The typical roulette wheel selection is adopted to select chromosomes for crossover. Chromosomes with better fitness are more likely to be selected. The concrete process is: first, calculate the cumulative fitness of each chromosome as:

$$c_t=\sum_{l=1}^{t-1}\frac{f(M_l)}{\sum_{q=1}^{NP} f(M_q)},\ i=1,2,...,NP; \tag{11}$$

second, randomly generate a number $r\in[0,1]$ and select the chromosome t if it satisfies $c_{t-1}<r\le c_t$; In the end, repeat the second step until NP chromosomes are selected.

2) Crossover: After performing the selection operator, two chromosomes are selected as parents with a probability p_c to produce offspring. The above crossover operation is performed

until all pairs of parents that meet the crossover condition complete the operation. Suppose that $\boldsymbol{M}_a$ and $\boldsymbol{M}_b$ are selected for crossover. Two offspring are produced by swapping rows from the r+1th row to the Mth row between $\boldsymbol{M}_a$ and $\boldsymbol{M}_b$, where $r \in (0, M\text{-}1)$ is a random integer. According to the representation scheme, each row in the matrix of a chromosome represents a test form. Hence, the above crossover operator generates new chromosomes by swapping several test forms found in the two parental chromosomes, as shown in Fig. 1.

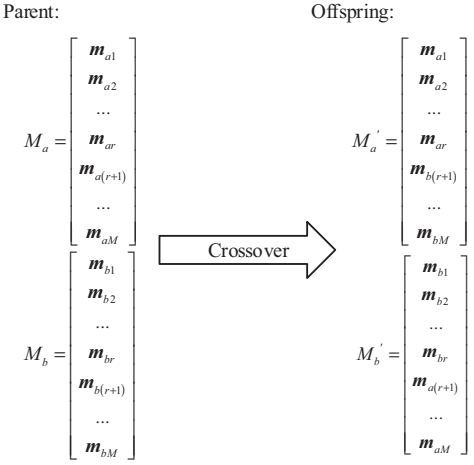

Fig. 1. Illustration of the crossover operator.

Note that the crossover operator may result in overlapping items between different test forms. Let $o_{ti} = \sum_{u=1}^{M} m_{ui}$ be the number of test forms in $\boldsymbol{M}_t$ that use the ith item (t = 1,2,...,NP, i=1,2,...,I). The following repair strategy is performed on $\boldsymbol{M}_t$ when o_{ti} exceeds the given threshold $h_{overlap}$. First, calculate the total cognitive diagnosis index (CDI) [15] of each test form in $\boldsymbol{M}_t$. The test form with a larger CDI usually has a better overall diagnostic accuracy. Hence, starting from the test form with the minimized CDI, substitute the ith item with one of the items that has not been used by any test forms in $\boldsymbol{M}_t$.

3) Mutation: A mutation operator is designed to increase the population diversity. For each offspring generated by the crossover, the mutation operator selects some genes from each test form with a probability p_m. The value of each selected gene is exchanged with the value of the other one. Note that the two genes must have different values because each gene in a chromosome is an indicator of whether using an item in a test and the number of items in each test form should remain constant. The mutation is only applied when the resulting test form still satisfies the overlapping threshold $h_{overlap}$.

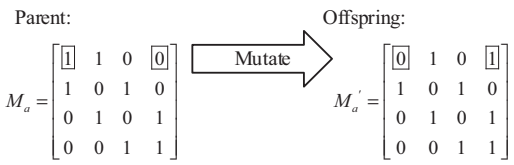

Fig. 2. Illustration of the mutation operator.

4) Elitism: The elitism operator is done to avoid loss of valid genes (i.e. items) so that better chromosomes are more likely to survive into the next generation. By doing so, the elitism strategy can help improve the convergence speed of GA. In detail, the elitism strategy first compares the fitness values of the offspring and its corresponding parental chromosomes. For each parental chromosome, if the corresponding offspring chromosome has a better fitness values, then enters next generation. Otherwise, if the parental chromosome has a better performance than the corresponding offspring chromosome, the parental chromosome has a probability p_e to replace the corresponding offspring chromosome to survive into the new population.

D. Overall Procedure

The flowchart of the proposed GA-based ATA method is shown in Fig. 1. As can be seen, the overall procedure contains the following steps:

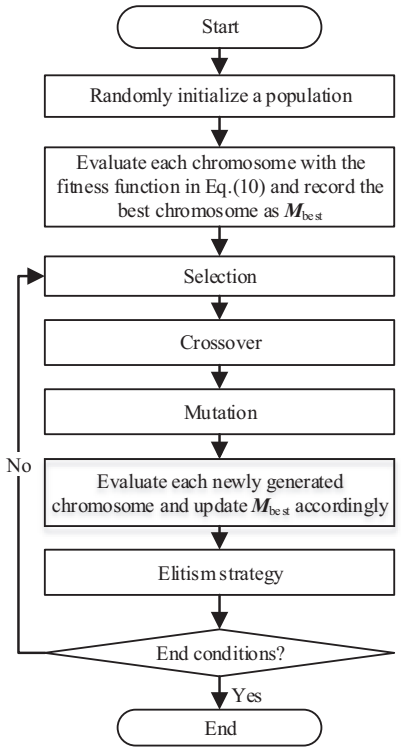

Fig. 3. The overall flowchart of the proposed GA-based ATA method.

1) Randomly initialize a population with *NP* chromosomes. Each chromosome is a candidate solution to the problem and each gene in the chromosome is an indicator of whether using the corresponding item in the corresponding test.

2) Evaluate each chromosome in the population with the fitness function in Eq. (10). Record the best-so-far chromosome as $\boldsymbol{M}_{\text{best}}$.

3) Apply the roulette wheel selection method to choose *NP* chromosomes for crossover.

4) Use the crossover operator described in Section III-C to produce *NP* new chromosomes.

5) For each new chromosome, use the mutation operator in Section III-C to randomly flip some genes.

6) Evaluate the offspring with the fitness function in Eq.(10) and update $\boldsymbol{M}_{\text{best}}$ if a better solution has been found.

7) Conduct the elitist strategy described in Section III-C to generate a new population.

8) If the end condition is satisfied (e.g., reaching the maximum number of generations or the quality of convergence), the evolution finishes and returns $\boldsymbol{M}_{\text{best}}$ as the result. Otherwise, return to Step 3) and proceed into the next generation.

IV. Experiments And Results

In this section, the proposed GA-based ATA method is validated on a synthetic item pool. The results are compared with a baseline method based on CDI to illustrate the advantage of the proposed method. Detailed experimental settings and results are given below.

A. Experimental Settings

To make a comprehensive study, experiments are conducted to build tests for all the three diagnosis purpose $F^{(1)}$, $F^{(2)}$, and $F^{(3)}$. The experiments require some necessary settings, including the item pool, prior distribution of $\boldsymbol{\alpha}$, practical constraints, and control parameters of the ATA methods.

1) Item Pool. The item pool is composed of 300 5-attibutes items. So the Q-matrix is a 300·5 0-1 matrix, which constrained to have 80 items measuring one attribute, 140 items measuring two attributes, and 80 items measuring three attributes, for an average of two attributes per item. Within each item, the attribute or attributes being measured are randomly determined. In the experiments, items are assumed to follow the reduced RUM, with π_i^* and r_{ik}^* are randomly generated in [0.20, 0.95] and [0.75, 0.95], respectively (i = 1,2,...,300, k = 1,2,...,5).

2) Prior Distribution of $\boldsymbol{\alpha}$. Two prior distributions of $\boldsymbol{\alpha}$ are considered. The first one is uninformative, where all the (2^5 = 32) attribute mastery patterns have equal prior probabilities, i.e., $P(\boldsymbol{\alpha})$ = 1/32. The other one is the multivariate normal distribution, which is considered more realistic than the former. The tetrachoric correlation of each pair of attributes is set to 0.5 and the percentage of masters on each attribute is set to 50%.

3) Constraints. During test construction, several constraints are considered in order to generate practical uniform test forms. Constraints include: (a) the required number of test forms is set as 5 (i.e., M = 5); (b) the test length is set as 40 (i.e., *Length* = 40) based on the capacity of 300-item pool; (c) the overlapping threshold is set as 1 (i.e., $h_{overlap}$ = 1), meaning that each item can only be used in one test form.

4) Set of Examinees. A set of virtual examinees is needed for evaluating the quality of a test form. To do so, a sample is derived from the prior distribution of $\boldsymbol{\alpha}$. During the optimization procedure of GA, the sample, namely the training set, contains 2,000 examinees. Then the test forms generated are evaluated using another sample of 2,000 examinees, namely the test set. Separating the training set and the test set is to avoid the unfair advantage stemmed from the fact that GA uses the training set in fitness evaluation. Note that all the results reported below are based on the test set.

5) Control Parameters of GA. The parameters for GA are defined as follow: NP = 20, p_c = 0.88, p_m = 0.1, p_e = 0.133, and the maximum iteration is set to 500. As a probabilistic algorithm, GA is run for 10 independent times and the statistics are reported for comparison.

6) Settings of CDI. The original CDI method is not designed to generate multiple test forms. In order to make it applicable to the problem of generating uniform test forms, a modified CDI method is proposed referring to the sequential methods [1]. The modified CDI method performs as follows: (a) use CDI to select items into one test form; (b) remove the selected items from the item pool; (c) if five test forms are already generated, the test forms are return as the result; otherwise, return to step (a) and start building another test form. As a deterministic algorithm, CDI is run for only one time and the statistics are reported for comparison.

B. Results & Comparison

The results of the proposed GA-based ATA method and the CDI method are shown in Table I.

TABLE I. Comparison Between GA and CDI

Objective	Prior	GA		CDI	*p*-value
		MEAN±SD	BEST		
$F^{(1)}$	UNI	1.546±0.002	1.606	1.473	0.001**
	MVN	1.635±0.000	1.672	1.558	0.000**
$F^{(2)}$	UNI	1.772±0.004	1.851	1.862	0.001**
	MVN	1.842±0.002	1.892	1.889	0.004*
$F^{(3)}$	UNI	1.753±0.002	1.816	1.633	0.000**
	MVN	1.763±0.001	1.804	1.614	0.000**

UNI = uniform distribution; MVN = multi-variant normal distribution;
* and ** indicate statistically significant difference at the level of 0.05 and 0.001, respectively.

Table 1 gives the experimental results based on three fitness functions for methods and conditions, including mean, standard deviation, and the best value. In order to be statistically sound, a simple two-side *t*-test is performed with the null hypothesis (H_0) that there is no difference between the optimization results (fitness values) of GA and CDI.

As can be seen from Table 1, the proposed method (GA) achieves better results in terms of $F^{(1)}$ and $F^{(3)}$ when compared with the baseline method (CDI). The *p*-values obtained using a two-side *t*-test indicate that the advantage of the proposed method is statistically significant ($p < 0.05$). Recall that in Section III-B, $F^{(1)}$ optimizes the overall classification error rate and $F^{(3)}$ optimizes the absolute distances between the actual and the target error rates. The significant advantage of GA implies that it is better at optimizing the overall quality of multiple test forms while maintaining similar test quality. However, the *t*-test for the results on $F^{(2)}$ also provides a significant *p*-values, which means that CDI obtains a lower maximum classification error rate across all the attributes than GA does. It is speculated that since $F^{(2)}$ is defined as the maximum attribute-level error rate, GA struggles to balance the test quality and thus has weaker performance on minimizing the maximum attribute-level error rate of all uniform test forms.

Overall, it can be concluded that GA is effective and efficient for generating uniform test forms such that each test form uses a different set of items and meets equivalent demand of test quality.

V. Conclusions

This paper proposes a GA-based ATA method for generating uniform test forms of CDMs. The proposed method is the first solution to uniform test form generation in the context of CDMs. Experiments on a synthetic item pool show the proposed method is effective and efficient.

In future, we will focus on improving the current method so that the solution quality and the computational speed are both enhanced. Two ways have been considered for doing so. One is to parallelize the algorithm using multi-population techniques on MPI platforms. The other is to handle the problem in a multi-objective framework and use multi-objective evolutionary algorithms to address the problem.

Besides improving the performance of the proposed method, we will also consider how to import and address the constraints of overlapping items so that the problem and its solution can be more practical.

References

[1] W.J. van der Linden and E. Boekkooi-Timminga, "A Maximin Model for Test Design with Practical Constraints," *Psychometrika*, vol. 54, pp. 237-247, 1989.

[2] E. Boekkooi-Timminga, "The Construction of Parallel Tests from Irt-BasedItemBanks," *J. EducationalStatistics*, vol.15, no.2, pp. 129-145, 1990.

[3] R.D. Armstrong, D. Jones, and I.-L. Wu, "An Automated Test Development of Parallel Tests from a Seed Test," *Psychometrika*, vol. 57, no. 2, pp. 271-288, June 1992.

[4] P.H. Chen, H.H. Chang, and H. Wu, "Item Selection for the Development of Parallel Forms from an IRT-Based Seed Test Using a Sampling and Classification Approach," *Educational and Psychological Measurement*, vol. 72, no. 6, pp. 933-953, 2012.

[5] P.H. Chen, "A sampling and Classification Item Selection Approach with Content Balancing," *Behavior research methods*, vol. 47, no. 1, pp. 98-106, 2015.

[6] J. Pereira, and M. Vila, "Variable Neighborhood Search Heuristics for a Test Assembly Design Problem," *Expert Systems with Applications*, vol. 42, no. 10, 2015.

[7] G.J. Hwang, H.C. Chu, P.Y. Yin, and J.Y. Lin, "An Innovative Parallel Test Sheet Composition Approach to Meet Multiple Assessment Criteria for National Tests," *Computers & Education*, vol. 51, no. 3, pp. 1058-1072.

[8] P. Borovska, "Solving the Travelling Salesman Problem in Parallel by Genetic Algorithm on Multicomputer Cluster," *Proc. Int'l Conf. Computer Systems and Technologies*, pp. II.11-1-II.11-6, 2006.

[9] K. He, L. Zheng, S. Dong, L. Tang, J. Wu, and C. Zheng, "PGO: A Parallel Computing Platform for Global Optimization Based on Genetic Algorithm," *Computers & Geosciences*, vol. 33, no. 3, pp. 357-366, 2007.

[10] A.J. Verschoor, "Genetic Algorithms for Automated Test Assembly," PhD dissertation, Univ. of Twente, 2007.

[11] K. Sun, Y. Chen, S. Tsai, and C. Cheng, "Creating IRT-Based Parallel Test Forms Using the Genetic Algorithm Method," *Applied Measurement in Education*, vol. 2, no. 21, pp. 141-161, 2008.

[12] P. Songmuang, and M. Ueno, "Bees Algorithm for Construction of Multiple Test Forms in E-Testing," *Learning Technologies, IEEE Transactions on*, vol. 4, no. 3, pp. 209-221, 2011.

[13] T.Y. Chang, and Y.F. Shiu, "Simultaneously Construct IRT-Based Parallel Tests Based on an Adapted CLONALG Algorithm," *Applied Intelligence*, vol. 36, no. 4, pp. 979-994, 2012.

[14] P.D. Nichols, S.F. Chipman, and R.L. Brennan, *Cognitively Diagnostic Assessment*, Hillsdale, NJ: Erlbaum, 1995.

[15] R.A. Henson, and J. Douglas, "Test Construction for Cognitive Diagnosis," *Applied Psychological Measurement*, vol. 29, no. 4, pp. 262-277, 2005.

[16] M.D. Finkelman, W. Kim, and L.A. Roussos, "Automated Test Assembly for Cognitive Diagnosis Models Using a Genetic Algorithm. Journal of Educational Measurement," *Journal of educational measurement*, vol. 46, no. 3, pp. 273-292, 2009.

[17] M.D. Finkelman, W. Kim, L.A. Roussos, and A. Verschoor, "A Binary Programming Approach to Automated Test Assembly for Cognitive Diagnosis Models," *Applied Psychological Measurement*, vol. 34, no. 5, pp. 310-326, 2010.

[18] A.A. Rupp, J.L. Templin, and R.A. *Henson, Diagnostic measurement: Theory, Methods, and Applications*, New York, NY: Guilford, 2010.

[19] A.C. George, "Investigating CDMs: Blending Theory with Practicality," Unpublished doctoral dissertation, Univ. of TU Dortmund, 2013.

[20] S.M. Hartz, "A Bayesian Framework for the Unified Model for Assessing Cognitive Abilities: Blending Theory with Practicality," Unpublished Doctoral Dissertation, University of Illinois at Urbana-Champaign, 2002.

[21] L.A. Roussos, L.V. DiBello, W. Stout, S.M. Hartz, R.A. Henson, and J.L. Templin, "The Fusion Model Skills Diagnosis System," in Cognitive Diagnostic Assessment for Education: Theory and Applications, J. P. Leighton & M. J. Gierl, Eds. Cambridge, UK: Cambridge University Press, 2007, pp. 275-318.

[22] R. Henson, J. Templin, and J. Douglas, "Using Efficient Model Based Sum-Scores for Conducting Skills Diagnoses," *Journal of Educational Measurement*, vol. 44, no. .4, pp: 361-376, 2007.

[23] J.H. Holland, "Genetic Algorithms," *Scientific American*, vol.267, no.1, pp. 44-50, 1992.