Three-Element Item Selection Procedures for Multiple Forms Assembly: An Item Matching Approach

Applied Psychological Measurement
2016, Vol. 40(2) 114–127
© The Author(s) 2015
Reprints and permissions:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/0146621615605307
apm.sagepub.com

Pei-Hua Chen¹

Abstract

In educational measurement, building multiple alternate test forms is usually highly desirable in practice. Test assembly problems are classified as NP-hard (Non-deterministic Polynomial-time hard) problems, and their computation time increases exponentially as the problem size grows. The purpose of this study was to construct parallel forms item-by-item based on a seed test, using two proposed item selection heuristic methods (random item sequence assigned to maximum distance form with content balance [R-MD-B] and random item sequence assigned to random form with content balancing [R-R-B]) incorporating a flexible content balancing method. The results were compared with two modified versions of the Armstrong, Jones, and Wu heuristics (random item sequence assigned to maximum distance form with Armstrong's replacement [R-MD-R] and random item sequence assigned to random form with Armstrong's replacement [R-R-R]) and two models incorporating mixed integer programming (MIP) methods. The results showed that the R-MD-B and R-R-B methods yielded results comparable with the two MIP methods with less computation time in terms of the item information functions, test characteristic curves, and content coverage. The results also showed that R-MD-B and R-R-B can be extended to handle test assembly constraints in practical settings.

Keywords

test assembly, item response theory, parallel forms, heuristic algorithm, seed test

Introduction

Developers of educational assessments and licensure tests are usually required to provide multiple equivalent forms to accommodate the need for administrations at different locations and time slots. The use of automated test assembly procedures reduces the workload of test developers and ensures the quality of parallel test forms. In the past two decades, various researchers have successfully implemented optimization-based automated test assembly techniques such as mixed integer programming (MIP; Cor, Alves, & Gierl, 2009; Finkelman, Kim, Roussos, &

¹National Chiao Tung University, Hsinchu, Taiwan

Corresponding Author:

Pei-Hua Chen, Department of Management Science, National Chiao Tung University, 1001 University Road, Hsinchu 300, Taiwan, R.O.C.

Email: paulachen@mail.nctu.edu.tw

Verschoor, 2010; Sanders & Verschoor, 1998; van der Linden & Adema, 1998) and enumerative heuristics (Armstrong, Jones, & Wu, 1992; Luecht, 1998; Swanson & Stocking, 1993).

The MIP method is widely used in current test assembly practice mainly because commercial software packages (usually called solvers) such as CPLEX (IBM, 2012) are available and provide optimal solutions. However, multiple parallel test construction problems are classified as “NP-hard” (Non-deterministic Polynomial-time hard). It is well known that the computation time required for NP-hard problems increases exponentially with problem size. Some problems usually remain unsolvable as the item pool size and the number of constructed forms increase. Even if the problem size is small, some problem types, such as with the minimax model, are known to be difficult to solve. For example, Brusco, Köhn, and Steinley (2013) reported that with CPLEX they had trouble finding a global optimal solution for their minimax bin-packing problem within a reasonable time frame. In test assembly problem formulations, the minimax model is commonly used, which minimizes the largest deviations from the target form in terms of test information functions (TIFs) or inter-item distances. When using the MIP method presents difficulty in finding an optimal solution or finding a feasible solution, developing a heuristics method becomes necessary.

The enumerative heuristics method has the advantage of providing a pretty good option when an optimal solution cannot be found in realistic time. For example, the Weighted Deviation Model (Swanson & Stocking, 1993) and the Normalized Weighted Absolute Deviation Heuristic (Luecht, 1998) have been widely used in various operational programs. Other types of heuristics have also been used to deal with NP-hard problems, including simulated annealing (Brusco et al., 2013; van der Linden, Veldkamp, & Carlson, 2004), tabu search (Belov & Armstrong, 2005; Hwang, Chu, Yin, & Lin, 2008), and genetic algorithms (Finkelmann, Kim, & Roussos, 2009; Sun, Chen, Tsai, & Cheng, 2008).

Armstrong et al. (1992) attempted to generate parallel tests by matching items in the item bank to a seed test (also called a reference form) based on their inter-item distances on a two-dimensional item parameter space, item by item. Armstrong et al. first used the network flow programming method to select items from the item bank to match with each item in the seed test. Then, they created a two-step heuristic that assigns items to create initial forms and then interchanges items among forms to maintain form-to-form equivalence and satisfy content requirements. The results were found to be within a 1% deviation from the optimal solution, and they only required less than 2 min to obtain (Armstrong & Jones, 1992).

Clearly, the final solution of Armstrong et al. (1992) was not confirmed as optimal, and it makes sense to use heuristics to solve the problem in the first place. Armstrong et al.’s combined method and the MIP method always yield the same solution unless the item pool or problem constraints are changed. Yet, testing programs usually require producing different forms every year, so sometimes finding the same set of items every time may not be desirable. And this heuristic approach only interchanges items among forms to satisfy the content constraints. It is possible to encounter a situation in which there are no eligible items to exchange among the forms.

To address such issues, the present study proposed two item selection heuristics and incorporated them with content balancing techniques to assemble multiple parallel forms of a seed test. The two newly developed heuristics were compared with two modified versions of heuristics developed by Armstrong et al. (1992), who used the MIP method to assemble six 30-item forms with four content groups from a 510-item pool, reporting that the computation time was many hours. They performed the study more than 20 years ago, and computation speed has improved considerably since then. It was, therefore, worthwhile to execute an MIP version to see whether the computation time would be acceptable in view of current computation speeds.

Item Matching Studies

As their matching criterion, most test assembly studies have chosen to use target TIFs in item response theory (IRT) applications (Cor et al., 2009; Sun et al., 2008; Theunissen, 1985; van der Linden & Boekkooi-Timminga, 1989) or use test reliability (Adema & van der Linden, 1989) in classical test theory applications. Tests that match based on collective indices such as TIF or test reliability may not be presumed to exhibit stable, similar properties any more than can those based on item matching. Tests built based on matching item parameters directly capture the main properties of the items in the seed test, thereby ensuring the satisfaction of all cumulative indices such as TIFs and test characteristic curves.

Gulliksen (1987) was the first to propose a matched random subtests method for splitting tests into parallel halves. This method matches each seed test item based on a group of adjacent items (called a subset) in a two-dimensional item parameter space (item difficulty and discrimination), assigning adjacent items randomly to produce parallel test halves. Van der Linden and Boekkooi-Timminga (1988) used the concept of item-to-target distance and reformulated Gulliksen's matched random subtests method into an MIP model to split a test into two or more parallel halves tests, using software called LANDO. Following the same concept, Ariel, Veldkamp, and van der Linden (2004) divided a master pool into four smaller non-overlapping pools with similar item parameters and content attributes for use as rotating item pools in computerized adaptive testing (CAT).

Armstrong et al. (1992) proposed a two-step approach that used the item-to-target measure to construct T parallel N -item forms based on a seed test. The first step uses network flow programming to find T items in the item bank matched with each of the items on the seed test such that the sum of $T \times N$ item-to-target distances is minimized. The second step is to create T parallel tests with the smallest total sum of item-to-target distances of assembled forms. Because the second step involves an NP-hard problem, Armstrong et al. created an item selection heuristic that assigns T sorted items with item-to-target distances in ascending order one after another to the T sorted forms, with the current sum of the distance deviations from the seed test in descending order. Because the item selection heuristic described above only considers the item statistics in building initial forms, there must follow an item interchange procedure among the initial T forms to balance the content constraint requirement. A detailed description of the item interchange procedure can be found in the appendix of Armstrong et al.

As mentioned, the heuristic of Armstrong et al. (1992) exchanged items among forms to achieve the content coverage requirement. However, the item interchange process may become an endless loop if no item can be found to exchange among these forms to satisfy the content category constraints. One solution is to select items based on their statistical and non-statistical constraints at the same time by combining the item selection heuristic and content balancing techniques.

Content Balancing Techniques

Most content balancing heuristic methods were originally developed primarily for CAT (S.-Y. Chen & Ankenman, 2004; Cheng & Chang, 2009; Cheng, Chang, & Yi, 2007; Leung, Chang, & Hau, 2003). Flexible content balancing methods have been developed to control the lower and upper bounds of content constraints. Some of the methods described above were developed for specific CAT item selection circumstances and may not be applicable in the multiple-form assembly context.

P.-H. Chen (2006, 2014) incorporated a flexible content balancing approach for the Cell Only method to select items based on a seed test. Under P.-H. Chen's approach, items were

chosen to satisfy the lower bounds of every content area first, and then remaining items were selected based on the abundance in the item pool without exceeding the upper bounds and test length constraints. The rationale used for P.-H. Chen's content balancing approach was applied in the present study. When items were selected based on their item-to-target distances, consideration was given first to fulfilling the lower bounds of constraints for all content categories at the same time, without exceeding restrictions for the upper bounds and test length.

Problem Statement

The test assembly problem involved the creation of F parallel forms from the item pool by matching a seed test item-by-item. Assembled test forms needed to satisfy non-statistical constraints such as test length and lower and upper bounds of content categories. No form-to-form overlap was allowed.

Inter-Item Distance Matrix Creation

The weighted item-to-target measure (d_{it}) based on a three-parameter logistic IRT model is defined as follows, based on a three-dimensional scatterplot with the a parameter on the x -axis, the b parameter on the y -axis, and the c parameter on the z -axis. For each item i in the pool, it is possible to compute the Euclidean inter-item measure d_{it} for each of the target items t , which can be written as $d_{it} = \sqrt{w_a(a_i - a_t)^2 + w_b(b_i - b_t)^2 + w_c(c_i - c_t)^2}$, where a_i , b_i , and c_i represent the a , b , and c parameters for an item i in the item pool. Similarly, a_t , b_t , and c_t represent the a , b , and c parameters for a target item t , and w_a , w_b , and w_c are positive values and represent weights for item parameters a , b , and c , respectively, where $w_a + w_b + w_c = 1$. Practitioners could choose different weights for each parameter based on the relative importance of item parameters.

Before implementing the four item selection methods, a sorted $T \times I$ item-to-target distance matrix with test length T and item pool size I was required. The values of item-to-target distance d_{it} at row t could be sorted from the smallest to the largest values to create a sorted $T \times I$ item-to-target distance matrix with ascending order based on the value of d_{it} per row.

Six Item Selection Methods

For the present study, four inter-item-distance item selection methods for multiple forms test assembly were introduced: (a) random item sequence assigned to maximum distance form with content balance (R-MD-B), (b) random item sequence assigned to random form with content balancing (R-R-B), (c) random item sequence assigned to random form with Armstrong's replacement (R-R-R), and (d) random item sequence assigned to maximum distance form with Armstrong's replacement (R-MD-R). These four methods were then compared with two MIP methods. One used the weighted sum of the inter-item distance as the objective function (IM MIP method) and the other used the sum of deviations of TIFs from the reference form as the objective function (TIF MIP method).

Heuristics Method 1: R-MD-B

This method randomly picks a row t from T rows. Then a higher priority is given to an item with a smaller value of d_{it} in row t when assigning items to the forms. The sum of the total inter-item distances between the target form and form f is written as $sumd_f = \sum_{i=1}^T d_{it} \times x_i$, where $x_i = 1$ if item i is selected and $x_i = 0$ otherwise. To reduce the form-to-form differences in

the sum of the total inter-item distances ($sumd_f$), a higher priority is given to the form f with the larger $sumd_f$ when assigning item i .

Before an item i within content area j is assigned to a form, it is necessary to calculate the accumulated number of items for each form in content area j and compare the values with the lower and upper bounds of content area j . The basic rule is that, during the item assignment process, the required lower bounds of content area j in all the F forms have to be fulfilled first without exceeding the upper bounds. The candidate item i within content area j will only be assigned if not all of the F forms have reached the upper bounds of content area j . Otherwise, another item i' with the next smallest value of $d_{it'}$ and in content area j' will become the next candidate item, and it will be evaluated based on the assignment rules described above, where $i \neq i'$ and $j \neq j'$. The item selection process described above will be repeated until all T items have been selected for each form.

Heuristics Method 2: R-R-B

This method is very similar to Method 1. The only difference is that items are assigned randomly to one of the F forms as long as adding an item satisfies the content area requirement during the item assignment process. This process does not control the form-to-form differences in the sum of inter-item distances ($sumd_f$).

Heuristics Method 3: R-R-R

This method also uses the sorted $T \times I$ inter-item distance matrix with ascending order based on the value of d_{it} for each target item t at row t , and it uses a two-step item selection process. For the first step, it randomly picks a target item t and finds the F items with the F smallest inter-item distances from the target item t at row t of the $T \times I$ inter-item distance matrix, and it assigns them randomly to F forms. The first step repeats until all T items in each of the F forms have been chosen.

The second step uses the item interchange rule proposed by Armstrong et al. (1992) to satisfy the content requirements. A detailed description of the item interchange rule can be found in Armstrong et al.

Heuristics Method 4: R-MD-R

While Method 3 finds the items with the minimum distance from the target items during the first step of the item selection process, items are randomly assigned to F forms, and the form-to-form differences in the sum of inter-item distances ($sumd_f$) are not controlled. Method 4 is almost the same as Method 3, except that Method 4 takes the value of $sumd_f$ into consideration during the process of assigning items to forms, giving a higher priority to assigning items with a small d_{it} value to the form f with the larger $sumd_f$ value.

Because of the item interchange procedure between forms at the second stage, the form-to-form differences in the sum of inter-item distances ($sumd_f$) of the resulting forms may not be as small as in the initial forms.

Method 5: Item Matching with Mixed Integer Programming (IM MIP)

In this model, forms are selected simultaneously, not one after another. The total sum of the inter-item distances d_{it} between an item i in the item pool and a target item t on the target test is used as the objective function. The model can be written as

$$\text{Minimize } \sum_{f=1}^F \sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf}, \quad (1)$$

subject to

$$\sum_{f=1}^F x_{itf} \leq 1, \text{ for all } t \text{ and all } f, \quad (2)$$

$$\sum_{t=1}^T \sum_{i=1}^I g_{ic}x_{itf} \leq UB_c, \text{ for all } c \text{ and } f, \quad (3)$$

$$\sum_{t=1}^T \sum_{i=1}^I g_{ic}x_{itf} \geq LB_c, \text{ for all } c \text{ and all } f, \quad (4)$$

$$\sum_{i=1}^I \sum_{t=1}^T x_{itf} = T, \text{ for all } f, \quad (5)$$

where x_{itf} is equal to 1 if item i in the pool is matched to item t on the seed test and assigned to form f , and x_{itf} is equal to 0 otherwise. Here g_{ic} equals 1 if item i belongs to content category c , and it equals 0 otherwise. UB_c and LB_c refer to the upper bound and lower bound, respectively, of content category c . The constraint in Equation 2 means that no item overlap between the two forms is allowed. The content category requirements are modeled by the constraints in Equations 3 and 4. The constraint on the test length is expressed by Equation 5.

Equation 1 does not control the form-to-form differences in the sum of inter-item distances and, therefore, is not a good choice for the objective function. To control the maximum value of the form-to-form differences in the sum of inter-item distances ($Msumd_f$), the objective function can be modified by adding an element called $Msumf$ while adding two other constraints. The final objective function (Equation 6) and two additional constraints can be expressed as follows:

$$\text{Minimize } Msumd_f + \sum_{f=1}^F \sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf}, \quad (6)$$

subject to

$$\sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf_1} - \sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf_2} \leq Msumd_f, \text{ for all } f_1 < f_2, \quad (7)$$

$$\sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf_1} - \sum_{t=1}^T \sum_{i=1}^I d_{it}x_{itf_2} \geq -Msumd_f, \text{ for all } f_1 < f_2. \quad (8)$$

The added element of $Msumd_f$ in the objective function along with Equations 7 and 8 minimizes the maximum value of the form-to-form differences in the sum of inter-item distances ($Msumd_f$), and, therefore, the objective function of Equation 6 becomes a minimax model, which is more difficult to solve than Equation 1.

Method 6: Test Information Function with Mixed Integer Programming (TIF MIP)

In practice, the TIF is usually used to determine the statistical equivalence in test assembly. Therefore, this study implemented an MIP model, which minimizes the form-to-form largest deviations from the target TIF and also the form-to-form TIF differences. The model can be written as

$$\text{Minimize } D + y, \quad (9)$$

subject to

$$\sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if} - TI(\theta_k) \leq y, \quad (10)$$

$$\sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if} - TI(\theta_k) \geq -y, \quad (11)$$

$$\sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if_1} - \sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if_2} \leq D, \text{ for all } f_1 < f_2, \quad (12)$$

$$\sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if_1} - \sum_{f=1}^F \sum_{k=1}^K \sum_{i=1}^I I_i(\theta_k) x_{if_2} \geq -D, \text{ for all } f_1 < f_2, \quad (13)$$

where $I_i(\theta_k)$ is the item information function of item i at ability level θ_k , x_{if} equals 1 if item i is assigned to form f , and x_{if} is equal to 0 otherwise.

In Equations 9, 10, and 11, y represents the largest deviations from the target TIF value. D in the objective function, along with Equations 12 and 13, controls the maximum value of the form-to-form TIF differences (D). Notice that the TIF MIP method is also a minimax model, because it contains both y and D in the objective function.

Simulation Studies

Three simulation studies were conducted. Study 1 used a real item pool with simple test specifications. Study 2 used simulated pools to investigate the performance of each of the item-selection methods under various test assembly conditions. Study 3 focused on examining the item selection methods to deal with practical test assembly constraints.

Study 1: Real item pool

A pre-calibrated real item pool composed of 540 items and three content areas was used. Among the 540 items, Content Area 1 contained 216 items, and Content Areas 2 and 3 each contained 162 items. A 40-item reference form was used as the target test. The lower and upper bounds for the content requirements were set to be between 14 and 18 items for Content Area 1 and between 10 and 14 items for each of the Content Areas 2 and 3. The actual size of the item pool was 500, because the reference form was selected from the item pool.

Item selection methods. Each of the six item selection methods introduced earlier was used to assemble five parallel forms to match the reference form, with flexible content constraints. When calculating the inter-item distances, unweighted and weighted versions were implemented. After examination of the TIF and test characteristics curves (TCCs) resulting from the

unweighted and weighted versions, these weights ($w_a = .5, w_b = .25, w_c = .25$) were used to represent the relative importance of a parameters to the information function. Each of the four heuristics methods was run repeatedly to obtain 100 successful trials. The largest distance value among the selected forms for each trial was calculated. The trial whose largest distance value among the forms was the smallest among the 100 successful trials was chosen and reported. The TIF MIP method was set up to match the TIF of the reference form at 13 ability points ($\pm 3, \pm 2.5, \pm 2, \pm 1.5, \pm 1, \pm .5, 0$). The results from using the IM MIP and the TIF MIP methods were obtained using CPLEX 12.5 (IBM, 2012).

Evaluation criteria. Two evaluation criteria were used: the positive and negative deviations from the target form of the TIFs and of the test characteristic curves. Content coverage was also checked to see whether all content categories fell within the upper and lower bounds of the content constraints. Because the Armstrong et al. (1992) replacement approaches (R-R-R and R-MD-R) satisfy the non-statistical constraints by swapping items between forms, it is possible that these two heuristics may occasionally not find any solution and become a failed trial. A trial was defined as *failed* when no solution was found after looping indefinitely 1,000 times. A failure rate index was defined as the number of failed trials divided by the sum of the number of failed trials and the 100 successful trials.

Results. The R-R-B and R-MD-B methods took 7.094 and 8.769 s to obtain 100 results using an Intel i-7 3700M CPU with 8 GB RAM memory, respectively. As suspected, the two methods with the Armstrong et al. (1992) replacement approaches (R-R-R and R-MD-R) occasionally encountered endless-loop problems, and their failure rates were 35.06% (R-R-R) and 34.64% (R-MD-R). The IM MIP method with Equation 6 as the objective function took 22.81 s to obtain an optimal solution with CPLEX 12.5 (IBM, 2012), whereas the method with Equation 1 as the objective function took only 10.73 s. The TIF MIP method could not find an optimal solution over a span of 30 min when it was run with default parameter settings. The program ran for more than 34 hr until CPLEX stopped, reporting it was out of memory. After various parameters in CPLEX were tuned, only 8 min and 37 s were required to obtain a feasible solution close to the 0.5 unit gap from the optimal solution. Even when the maximum value of the form-to-form differences (D) was dropped from the objective function in Equation 9, the CPLEX solver still could not prove to be optimal within 30 min. With regard to the content constraints, the number of items per content area in all assembled forms was within the required upper and lower bounds.

Figure 1 shows the deviations of the TIFs (Forms A-E) from the target form resulting from the six item selection methods (IM MIP, TIF MIP, R-MD-B, R-R-B, R-R-R, and R-MD-R). All of the methods yielded deviations from the target form of ± 1 unit of information function along the ability scale. As expected, the TIF MIP method yielded the smallest deviations, within ± 0.5 unit of information function, because it was set up to minimize the sum of deviations of the TIF from the reference form.

Figure 2 shows the deviations of the five test characteristic curves (Forms A-E) from the target form resulting from the six methods. All of the four heuristic methods and the IM MIP method yielded deviations from the target form within 1 unit of test characteristic curve. Only the TIF MIP method yielded positive deviations from the target form of more than 1 unit at ability levels less than -1 .

Study 2: Simulated pools

The goal of the simulated pools study was to further investigate the performance of each of the item selection methods under various conditions: item pool sizes, non-statistical constraints, and number of assembled forms.

Figure 1. The deviations of the test information functions from the target form.

Note. MIP = mixed integer programming; R-MD-R = random item sequence assigned to maximum distance form with Armstrong's replacement; R-R-R = random item sequence assigned to random form with Armstrong's replacement; R-MD-B = random item sequence assigned to maximum distance form with content balance; R-R-B = random item sequence assigned to random form with content balancing; TIF = test information function.

Item parameters. The reference test and all the item parameters in each pool were generated as follows. Item discrimination parameters (a) were generated from a normal distribution with mean equal to 1 and standard deviation equal to 0.3. Item difficulty parameters (b) were generated from a normal distribution with mean equal to 0 and standard deviation equal to 1. Because item discrimination parameters and item difficulty parameters usually have a positive relationship (Chang, Qian, & Ying, 2001), a and b parameters were created to have a .4 correlation coefficient. The guessing parameters c were generated from a uniform distribution between 0 and 0.3.

Figure 2. The deviations of the test characteristic curves from the target form.

Note. MIP = mixed integer programming; R-MD-R = random item sequence assigned to maximum distance form with Armstrong's replacement; R-R-R = random item sequence assigned to random form with Armstrong's replacement; R-MD-B = random item sequence assigned to maximum distance form with content balance; R-R-B = random item sequence assigned to random form with content balancing; TCC = test characteristics curve.

Pool size, test length, and number of forms. Four item pool sizes were considered: 500, 1,000, 3,000, and 6,000. A 30-item reference test was used for the 500-, 1,000-, and 3,000-item pools. Both a 30-item reference test and a 60-item reference test were used for the 6,000-item pool. The number of required forms for test assembly was set up to be 5 and 10 forms.

Non-statistical constraints. The simulated items included three content areas and three cognitive categories. Two conditions of the content constraints were considered: (a) content areas with lower and upper bounds and (b) two-dimensional non-statistical constraints including both content areas and cognitive categories with upper and lower bounds.

Item selection methods. The four heuristic methods and the IM MIP method described earlier in the real item pool were used. The TIF MIP method was excluded in the simulation pool study, because it had shown difficulty finding an optimal solution within a reasonable time with the real item pool.

Evaluation criteria. Because the four methods and the IM MIP methods produced similar results in terms of deviations from the reference form by their TIF and test characteristic curves, the simulation pool only reported the required total running time of each of the item selection methods.

Results. In terms of running time, the R-MD-B and R-R-B methods performed the best among the five methods under all conditions. When only a content constraint was involved, the R-R-R and R-MD-R methods occasionally ran into endless loop in some trials, so they took more time to obtain 100 successful trials. The failure rates of the R-R-R and R-MD-R methods ranged from 27.01% to 93.79%.

When cognitive constraints were added, all of the R-R-R and R-MD-R methods resulted in endless loops and could not find any solutions. Although the IM MIP method could obtain an optimal solution, it generally took much more time than did the other heuristics methods. When the reference test length was 60 and the required number of form was 10, the IM MIP method could not even find an initial solution and was forced to stop after 80 min and 40 min, respectively, for the content constraint only and for the condition with cognitive constraint, because CPLEX ran out of memory while solving the model.

Study 3: Adding enemy item and response time constraints in simulated pools

To implement these test assembly methods under more practical conditions, in Study 3 two additional constraints of enemy item and response time were added on top of Study 2. Based on the results of Study 2, it is clear that the R-MD-R and R-R-R methods failed occasionally when both content and cognitive constraints were considered. Therefore, only the R-R-B, R-MD-B, and IM MIP methods were compared in Study 3.

Data generation for enemy items and response times. One set of enemy items with two or three members was randomly generated for every 100 items in the pool. For example, in the 500-item pool, five sets of enemy items containing two or three members were created. Response time data were generated using a log normal distribution with mean of 0.7 min and standard deviation of 0.3 min. The response time data were generated to have a .3 correlation coefficient with *b* parameters.

Non-statistical constraints. The non-statistical constraints in Study 3 included content areas, cognitive categories, enemy items, and response times. Study 3 used the same content and cognitive constraints as in Study 2. Enemy items were set up so that they could not coexist on the same form. For test length equal to 30, the lower and upper bounds of the response time constraint were 55 and 75 min. The lower and upper bounds of the response time for a test length of 60 were 110 and 150 min, respectively.

Results. The total running time results of Study 3 are very similar to the results of Study 2. In general, the running time for R-R-B and R-MD-B are almost the same. The IM MIP method took more time than the R-R-B and R-MD-B methods. When the reference test length was 60 and the required number of forms was 10, the solver was forced to stop after 44 min due to running out of memory with default parameter settings. Further adjustment of the parameter settings was required to obtain a solution after 2 hr and 53 min.

Discussion and Conclusions

In general, the resulting TIFs and test characteristic curves for the four heuristics methods indicate that the methods worked satisfactorily and that they produced solutions quite similar to the optimal solution found using the IM MIP method. Study 3 showed that the R-MD-B and R-R-B methods can be extended to handle practical constraints such as content areas, cognitive categories, enemy items, and response times.

Armstrong's heuristics have the merit of selecting items with smaller item-to-target distances to build initial forms. However, exchanging items only among the initial forms does not guarantee that there are always eligible items for interchange to satisfy the content constraints every time, especially when the item pool is not balanced in terms of item parameter distributions and content composition. To the contrary, the R-MD-B and R-R-B methods may not pick the item with the smallest item-to-target distance while concurrently considering the content constraints, so these two methods can always find a solution. In addition, the original heuristic method of Armstrong et al. (1992) does not consider randomly picking items in the target test to be matched during the item matching process, and therefore the resulting set of test forms will be the same all the time. The R-MD-R and R-MD-B methods can generate T different sets of solutions for a T -item seed test. Furthermore, the R-R-R and R-R-B methods use two random steps and can generate T^2 sets of solutions.

One of the primary advantages of the MIP method is its flexibility in solving various test assembly problems. The computation power available with today's personal computers could reduce computation time considerably. There are many factors that affect the MIP model's solving time, such as the size of the problem, the complexity of the objective function, and the model type (e.g., minimax), among others. For MIP test assembly problems with a larger item pool, more forms to assemble, and a longer test length, the computation time for such NP-hard problems increases exponentially as the problem size grows. In some cases, even a small MIP problem can be hard to solve and become intractable, such as the minimax model of the TIF MIP method. There may also be cases in which some problems may remain unsolvable within a practical period of time. Even if a solution can be found after further parameter adjustments, most test specialists are not familiar with the behind-the-scene mechanisms of the optimization solver, and, therefore, some extra effort and knowledge are needed. In such cases, developing alternative heuristics methods seems inevitable. In addition, solutions found by heuristic methods can be used as initial solutions for integer programming software to determine whether an initial solution found by a heuristic can be improved or even confirmed as an optimal solution.

Author's Note

An earlier version of this article was presented at the annual meeting of American Educational Research Association, Philadelphia, Pennsylvania, in 2014.

Acknowledgments

The author would like to thank the editor, associate editor, and two anonymous reviewers for their helpful comments on earlier versions of this article. Special thanks to Daniel Junglas for his helpful comments during the revision.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

- Adema, J. J., & van der Linden, W. J. (1989). Algorithms for computerized test construction using classical item parameters. *Journal of Educational Statistics*, 14, 279-290.
- Ariel, A., Veldkamp, B. P., & van der Linden, W. J. (2004). Constructing rotating item pools for constrained adaptive testing. *Journal of Educational Measurement*, 41, 345-359.
- Armstrong, R. D., & Jones, D. H. (1992). Polynomial algorithms for item matching. *Applied Psychological Measurement*, 16, 365-371. doi:10.1177/014662169201600408
- Armstrong, R. D., Jones, D. H., & Wu, I. L. (1992). An automated test development of parallel tests from a seed test. *Psychometrika*, 57, 271-288.
- Belov, D. I., & Armstrong, R. D. (2005). Monte Carlo test assembly for item pool analysis and extension. *Applied Psychological Measurement*, 29, 239-261.
- Brusco, M., Köhn, H., & Steinley, D. (2013). Exact and approximate methods for a one-dimensional minimax bin-packing problem. *Annals of Operations Research*, 206, 611-626. doi:10.1007/s10479-012-1175-5
- Chang, H.-H., Qian, J., & Ying, Z. (2001). a-Stratified multistage computerized adaptive testing with b blocking. *Applied Psychological Measurement*, 25, 333-341.
- Chen, P.-H. (2006, April). *IRT-based automated test assembly: A sampling and stratification approach with content balance*. Paper presented at the Annual Meeting of the National Council on Measurement in Education, San Francisco, CA.
- Chen, P.-H. (2014). A sampling and classification item selection approach with content balancing. *Behavior Research Methods*, 47, 98-106. doi:10.3758/s13428-014-0452-4
- Chen, S.-Y., & Ankenman, R. D. (2004). Effects of practical constraints on item selection rules at the early stages of Computerized Adaptive Testing. *Journal of Educational Measurement*, 41, 149-174. doi:10.1111/j.1745-3984.2004.tb01112.x
- Cheng, Y., & Chang, H. H. (2009). The maximum priority index method for severely constrained item selection in computerized adaptive testing. *British Journal of Mathematical and Statistical Psychology*, 62, 369-383. doi:10.1348/000711008x304376
- Cheng, Y., Chang, H.-H., & Yi, Q. (2007). Two-phase item selection procedure for flexible content balancing in CAT. *Applied Psychological Measurement*, 31, 467-482. doi:10.1177/0146621606292933
- Cor, K., Alves, C., & Gierl, M. J. (2009). Three applications of automated test assembly within a user-friendly modeling environment. *Practical Assessment, Research & Evaluation*, 14(14). Available from <http://pareonline.net/pdf/v14n14.pdf>
- Finkelman, M. D., Kim, W., & Roussos, L. A. (2009). Automated test assembly for cognitive diagnosis models using a genetic algorithm. *Journal of Educational Measurement*, 46, 273-292.
- Finkelman, M. D., Kim, W., Roussos, L. A., & Verschoor, A. (2010). A binary programming approach to automated test assembly for cognitive diagnosis models. *Applied Psychological Measurement*, 34, 310-326. doi:10.1177/0146621609344846
- Gulliksen, H. (1987). *Theory of mental tests*. Hillsdale, NJ: Lawrence Erlbaum.
- Hwang, G.-J., Chu, H.-C., Yin, P.-Y., & Lin, J.-Y. (2008). An innovative parallel test sheet composition approach to meet multiple assessment criteria for national tests. *Computers & Education*, 51, 1058-1072. doi:10.1016/j.compedu.2007.10.006
- IBM. (2012). IBM ILOG CPLEX (Version 12.5). Armonk, NY: Author.
- Leung, C. K., Chang, H. H., & Hau, K. T. (2003). Computerized adaptive testing: A comparison of three content balancing methods. *The Journal of Technology, Learning and Assessment*, 2(5). Available from <http://ejournals.bc.edu/ojs/index.php/jtla/article/view/1665/1507>
- Luecht, R. M. (1998). Computer-assisted test assembly using optimization heuristics. *Applied Psychological Measurement*, 22, 224-236.

- Sanders, P. F., & Verschoor, A. J. (1998). Parallel test construction using classical item parameters. *Applied Psychological Measurement, 22*, 212-223.
- Sun, K.-T., Chen, Y.-J., Tsai, S.-Y., & Cheng, C.-F. (2008). Creating IRT-based parallel test forms using the genetic algorithm method. *Applied Measurement in Education, 21*, 141-161.
- Swanson, L., & Stocking, M. L. (1993). A model and heuristic for solving very large item selection problems. *Applied Psychological Measurement, 17*, 151-166.
- Theunissen, T. (1985). Binary programming and test design. *Psychometrika, 50*, 411-420. doi: 10.1007/bf02296260
- van der Linden, W. J., & Adema, J. J. (1998). Simultaneous assembly of multiple test forms. *Journal of Educational Measurement, 35*, 185-198. doi:10.1111/j.1745-3984.1998.tb00533.x
- van der Linden, W. J., & Boekkooi-Timminga, E. (1988). A zero-one programming approach to Guiliksen's matched random subtests method. *Applied Psychological Measurement, 12*, 201-209. doi: 10.1177/014662168801200210
- van der Linden, W. J., & Boekkooi-Timminga, E. (1989). A maximin model for IRT-based test design with practical constraints. *Psychometrika, 54*, 237-247. doi:10.1007/bf02294518.1007/bf02294518
- van der Linden, W. J., Veldkamp, B. P., & Carlson, J. E. (2004). Optimizing balanced incomplete block designs for educational assessments. *Applied Psychological Measurement, 28*, 317-331.