GRADUATE MANAGEMENT ADMISSION COUNCIL

An Approach to Assembling Optimal Multistage Testing Modules on the Fly

Kyung T. Han and Fanmin Guo

GMAC® Research Reports • RR-13-01 • March 4, 2013

Abstract

This study presents a new adaptive multistage testing method that replaces the preassembled test module with a test module assembled on the fly after each stage. In this method, a test module for each stage is shaped to come as close as possible to the normal density function of the interim proficiency estimate and its standard error. We will hereafter call the new method 'multistage test by shaping' (MST-S) and refer to the traditional multistage test as MST by routing (MST-R). The new MST-S offers the advantages of both MST-R and computerized adaptive testing (CAT). With MST-S, the difficulty of a test module always centers on the latest interim proficiency estimate, which means it potentially can administer a test module that is more efficiently adapted to the individual compared with MST-R. Because test items are not necessarily limited to a certain stage but instead are available for use in any stage, the number of items required to implement MST-S can be much smaller than those required in MST-R. If desired, MST-S also can allow examinees to move back and forth within each module, as they can do now in MST-R. This study consists of a series of simulation studies that were conducted to evaluate the performance of MST-S in comparison with MST-R and CAT.

Introduction

With the emergence of item response theory and the rapid advancement of computer technology, computerized adaptive testing (CAT) is being used widely in a variety of testing applications across fields ranging from education to health and medicine. Unlike a test with a fixed form (for example, a paper-and-pencil exam), CAT assembles tests at the item level, achieving optimized measurement efficiency by administering each item that is the most relevant to each individual's proficiency level (Weiss, 1974). Numerous tools and techniques have been developed to implement CAT that meet important statistical and/or nonstatistical targets and constraints often including—but not limited to—measurement accuracy, item exposure rate, content balancing, test length, and item latency.

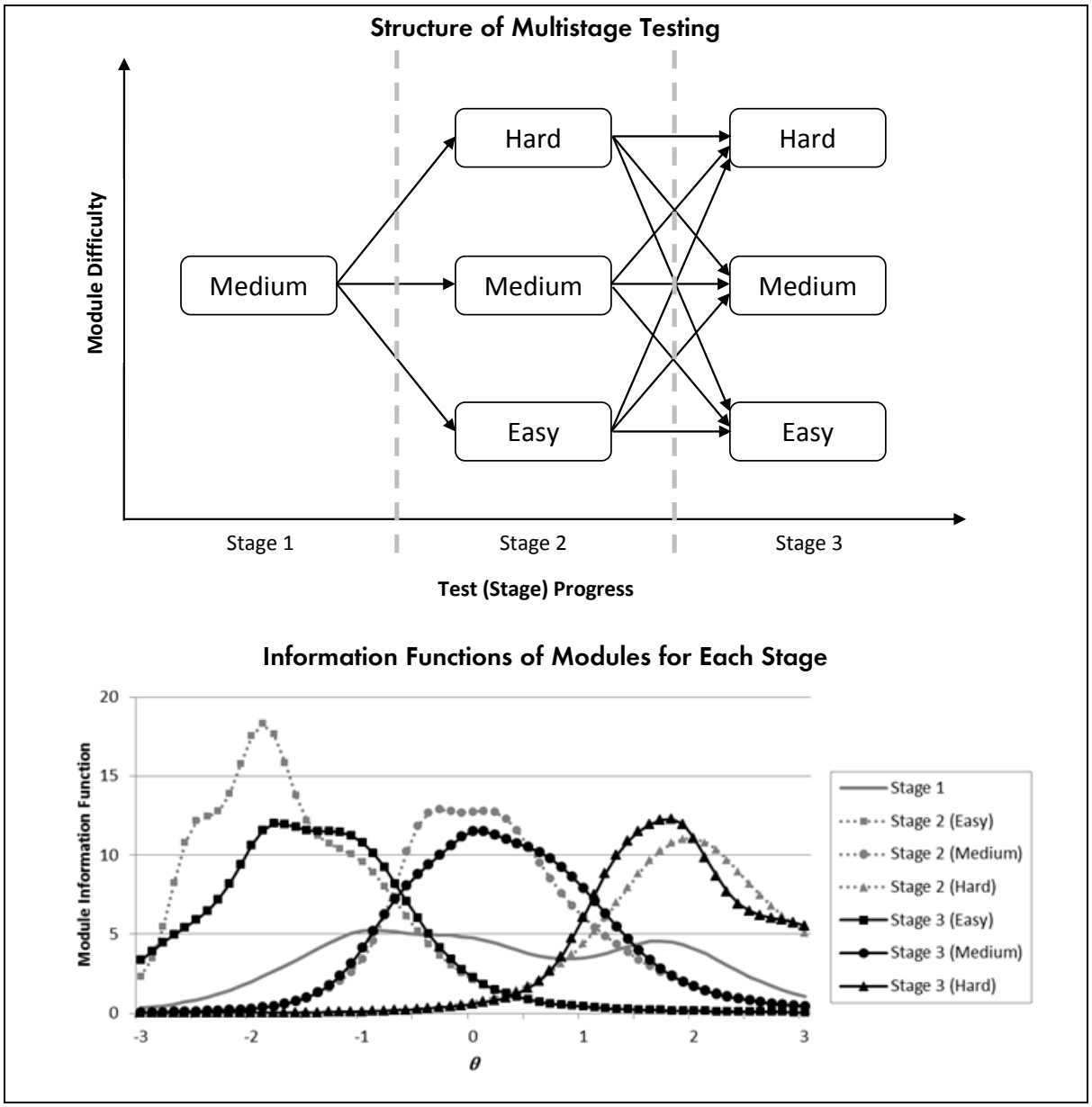
Multistage testing, or MST, was developed as an alternative to CAT for applications where it is preferable to administer a test at the level of a pre-assembled item set (i.e., module; Luecht & Nungester, 1998). One of the downsides of MST, however, is that the pre-assembled item sets may not be optimized to result in the best measurement efficiency at each stage (Lord, 1980; Zenisky, Hambleton, & Luecht, 2010). This study presents a new adaptive multistage testing method that assembles an optimized test module on the fly after each stage rather than relying on preassembled test modules.

Multistage Testing

As its name implies, a multistage test (MST) is divided into multiple stages and adaptively administered for each stage with a module whose difficulty level is the closest to examinee's expected proficiency.

Figure 1, for example, shows some typical structures of MST. In this example, the test was divided into three stages, with one module in the first stage and three modules in each of the second and third stages. Such a design often is referred to as the "1-3-3" module design (Luecht, Brumfield, & Breithaupt, 2006; Jodoin, Zenisky, & Hambleton, 2006). In this design, an examinee starts with the first stage, whose item module difficulty level is usually medium or average. After the first stage, the examinee is routed to one of the three preassembled item modules in the second stage based on his or her performance in the first stage. After completing the second stage, the examinee is again routed to one of the three item modules in the third stage. Thus, MST behaves essentially like a special case of CAT, which adaptively routs each test taker to one of several preassembled item groups based on his or her performance on the previously administered items. In the same respect, a typical CAT also can be regarded as a special case of MST, in which each stage consists of a single item and items are not tied to a single specific stage.

Figure 1 Illustrations of Multistage Testing



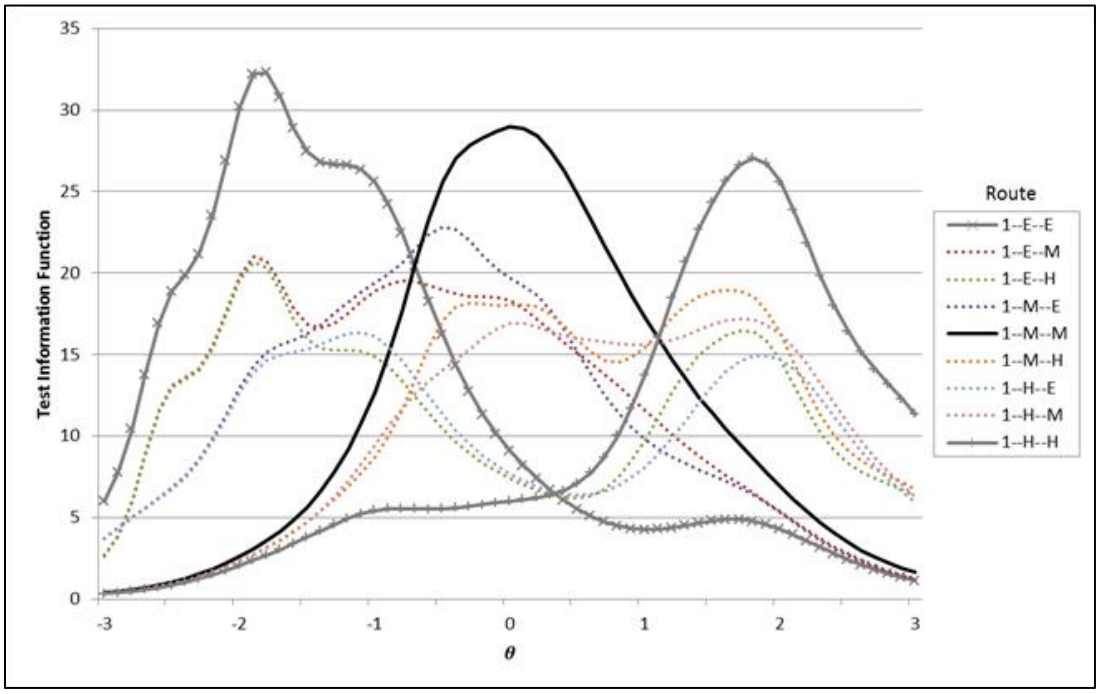
Administering a group of fixed items at one time rather than administering items individually could have some advantages depending on the test situation. For example, some tests consist of item sets, reading passages for example, that contain commonly shared content. In these instances, it would be appropriate to administer such an item set at one time as a module to avoid possible complications with item dependency and enemy management. Also, because the structure of stages, the placement of modules, and the composition of items within each module are almost always predetermined before the test administration, MST often offers more controls over the details of test specifications and properties. Another advantage of MST in comparison with CAT is that it places a smaller burden on client computers in terms of the item selection process. With MST, a client computer needs only to compute interim proficiency estimates after each stage instead of after each item. The computational workload for the selection algorithm is much simpler in MST, as well, because it considers only a handful of item modules as opposed to choosing from among hundreds of individual items. More important, examinees often prefer MST because it usually, if not always, allows them to move back and forth across items and change their initial responses within each module, unlike CAT, which prevents examinees from moving back once they submit their responses.

MST does have its downsides, however, such as a substantial tradeoff in the level of adaptability, which may eventually have a negative impact on the measurement efficiency. By increasing the number of stages, MST's adaptability could be improved; however, it would require considerably more items to build an MST with many stages compared with CAT for the simple reason that in MST, item modules are designed for use only in one specific stage. Item modules in one stage cannot be considered for selection in another even if they meet all other requirements.

Another major drawback of MST is the inconsistency in final test information function (TIF) within and/or across proficiency levels. When modules are preassembled in MST, the information function level across modules for each stage is usually controlled to be consistent regardless of examinee proficiency level. For example, as shown in the bottom of Figure 1, the information functions for modules in Stages 2 and 3 have similar shapes (except for the easy module for Stage 3, which shows higher information function in the lower proficiency area), differing only where the module information functions peak. Assembling item modules that have information functions with similar shapes usually requires use of sophisticated optimization techniques such as mixed integer programming (van der Linden, 2005; Melican, Breithaupt, & Zhang, 2010; Breithaupt, Ariel, & Hare, 2010; Zenisky et al., 2010). Constructing item modules with similar information function shapes for each stage is often considered important because it would help maintain consistency of the information function contributed by a selected item module regardless of the module choice at any stage of testing. There is no guarantee, however, that this will result in final TIFs that are consistent across examinees who were routed to different modules throughout the test. For example, Figure 2 shows the final TIFs from all possible routes and module combinations that were shown in Figure 1. For examinees whose proficiency level was -2.0 and who were routed to '1-E-E' (easy modules at the second and third stages after the first stage) modules, the final TIF was about 32.5. For examinees with the same proficiency level (= -2.0) who took different routes, for example, '1-E-M' or '1-M-E,' the final TIF was about 21 or lower. This is about a 30% difference in TIF for the same proficiency level, possibly large enough to raise a flag about test reliability control issues across examinees in some testing programs. Higher proficiency levels showed similar observations—compare '1-M-M' to '1-E-M,' '1-H-M,' '1-M-E,' or '1-M-H' at $\theta = 0$ and '1-H-H' to '1-E-H,' '1-M-H,' '1-H-E,' or '1-H-M' at $\theta = 2$. Although the choice of module difficulty tends to stay the same across stages for a majority of examinees (the solid curves in Figure 2), a significant number of examinees would still unavoidably end up with much lowers TIFs as a result of being routed to modules with different difficulty levels during the test (dot curves in Figure 2). Plus, it would still be common to observe substantial fluctuations in TIFs across proficiency levels even among the examinees who took the same route of modules. For example, in Figure 2, the TIF for examinees at the proficiency level of 1.0 who took the '1-M-M' route was less than 60% of the proficiency

level observed for examinees at $\theta = 0$ who took the same '1-M-M' route. It is thus apparent that the TIF and the standard errors of estimation in MST often seriously differ across examinees with different proficiency levels and also could substantially differ even across examinees with the same proficiency level if they were routed to different paths of test modules, which may introduce a huge challenge in controlling the test reliability across examinees.

Figure 2. Example of Inconsistent Test Information Function (TIF) Across Different Routes & Proficiency Levels



Another problem of MST occurs when an interim proficiency estimate is very close to the cut score. Depending which test module routing is decided for a following stage, there is a considerable likelihood that the module selected for the next stage will be less than optimal. This can be problematic especially when an interim proficiency estimate is unreliable and showing large standard errors of estimation during earlier stages of MST. Such a problem becomes even more serious when there are fewer modules at each stage and insufficient overlap between modules in term of item difficulty.

Individual test programs with differing MST designs often have completely different psychometric properties, so it is important to understand that the aforementioned advantages and disadvantages of MST do not necessarily generalize to all test programs that use MST. Despite the distinct advantages that MST offers, the need to use preassembled modules in many MST designs clearly makes it challenging to control TIFs.

MST by Shaping

The goal of this study was to present a new approach to MST that addresses the challenges of MST in controlling TIFs and finds ways to use items with improved adaptability while retaining the advantages of traditional MST designs, such as allowing examinees to move back and forth within a stage and increasing emphasis on nonstatistical specifications.

The proposed MST method does not select preassembled test modules. Instead, it assembles a test module 'on the fly' after each stage, using the following steps to assemble the new item module for the next stage:

1. Evaluate TIF and estimate an interim θ based on items administered thus far,
2. Evaluate the difference between the current TIF and target TIF for the next stage at the interim θ,
3. Construct a TIF mold, a new term to describe an ideal shape for the information function of the next item module (excluding previously administered modules) based on Step 2,
4. Shape an item module based on the mold in Step 3,
5. Administer the item module that was shaped in Step 4, and
6. Repeat Steps 1 to 5 until the last stage finishes.

Examples shown in Figure 3 illustrate these five steps. Step 1 of Figure 3 pictures a situation in which MST builds an item module for the third stage of a multistage test. The TIF and the interim θ estimate were computed after completion of the second test stage. In Step 2, the new MST method computes the difference between the current TIF (black solid curve) and the target TIF (gray dashed curve) that is centered on the interim θ estimate (red solid line). Test developers predetermine the target TIFs for each stage but the targets TIFs only dictate the shape of the TIF, not the location of the peak. In Step 3, the area difference in TIFs is directly translated into a TIF mold (shaded area) for the next stage. In Step 4, the new MST method selects a group of items for the purpose of shaping an item module with a TIF that resembles, as closely as possible, the mold created in Step 3. For the shaping step, the content balancing component is considered first. According to the test specification for each stage, the number of items needed for each content area is determined next. Module shaping, which involves iterative item selection processes, then begins, filling the item needs for each content area. The details of the module-shaping algorithm come next.

In typical CAT programs, the item selection algorithm looks for the best item based on the item selection criterion and then introduces a random factor to control exposure rate. For example, some CAT programs will use the maximum Fisher information criterion to choose an item that results in the highest Fisher information at the interim θ estimate, and then will apply the Sympson and Hetter (1985) method or conditional/unconditional multinomial methods (Stocking & Lewis, 1995, 1998), which ultimately introduce a huge random factor to control the probability of administering the selected items. For the exposure control in conventional MST applications, it is common for MST developers to build multiple equivalent panels—a set of modules with routing rules. Examinees are randomly assigned to one of the panels (Luecht et al., 2006), so the test exposure rate is controlled to be $1/k$ with k being the number of panels. The new MST method was developed with a built-in exposure control feature in the module-shaping algorithm. In opposite to typical CAT algorithms that first seek the best item and then introduce another random factor for exposure control, the new MST module-shaping algorithm begins with a random selection of items. According to the identified number of items needed for each content area in the next stage, the MST method then randomly draws eligible items from the item bank. Once the initial random drawing of items is finished, the iterative shaping routines begin. Each iteration of the module-shaping routine consists of following processes:

1. Assess the squared area difference between the current set of items (from the initial random drawing if it is the first iteration of the module shaping) and the TIF mold for next stage. The squared area difference can be expressed as

$$[I\ \theta\ ^{*} - \tau_{\theta_s}]^2\, d\theta, \qquad (1)$$

   where τ_{θ_s} is the TIF mold for stage s and $I\ \theta\ ^{*}$ is the TIF of the currently selected items.

2. For item i in the current set, randomly draw another item among the eligible items from the item bank. Replace item i with the new random draw and compute the squared area

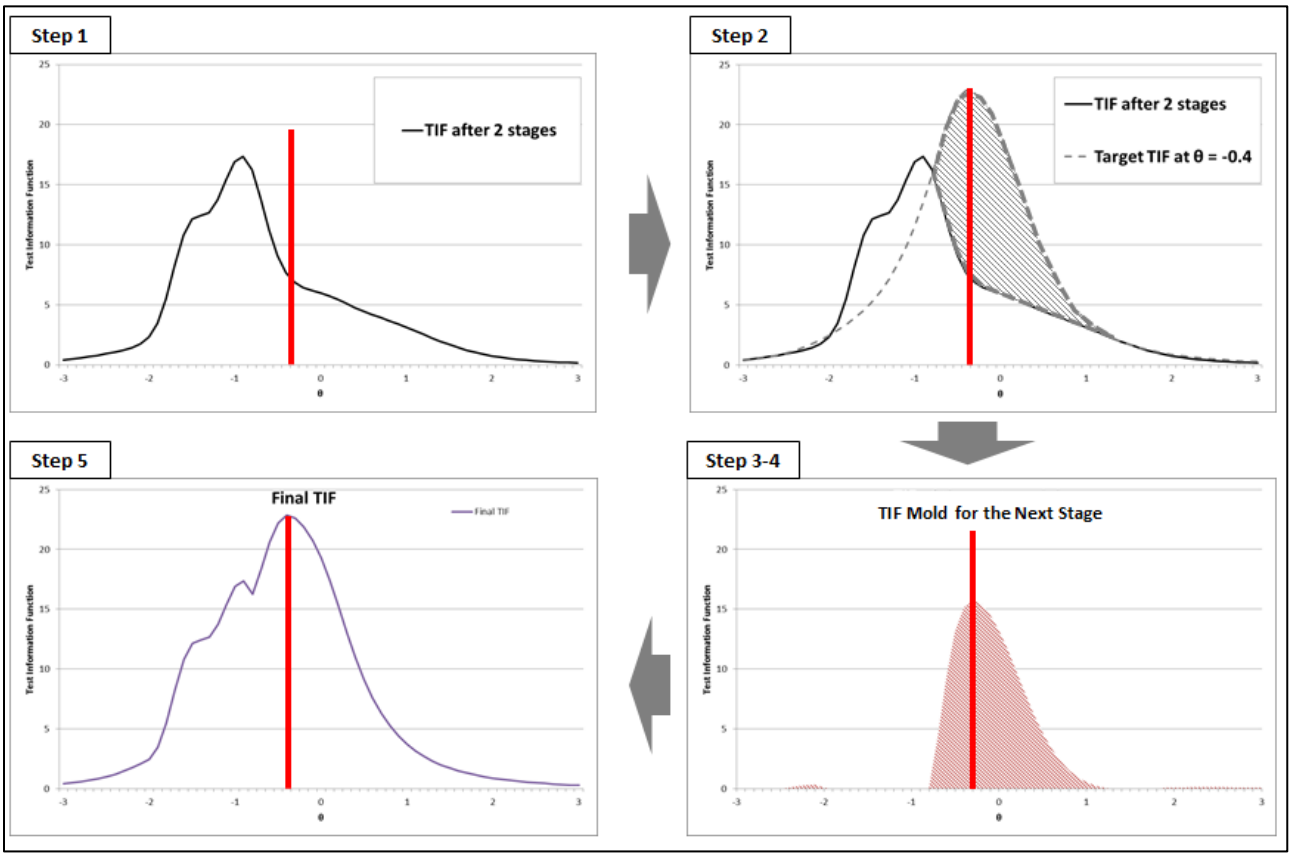
difference (Equation 1). If the squared area difference decreases with the replacement, keep the replacement. If the squared area difference stays the same or increases, discard the new random draw and retain the previous selection.

3. Repeat the process described in (2) for each of the currently selected items.
4. Iterate (1) to (3) until the number of iterations reaches the target.

Note that the shape of the module being built for the next stage comes closer to the TIF mold as the module-shaping process iterates. If the number of iterations for the module-shaping process is large, for example, as large as the number of eligible items in the item bank, then the shape of the finalized module for the next stage will likely be the one that is as close to the TIF mold as possible given the item bank. In this case, however, the random factor for the individual item selection would be minimized and so too would be the level of exposure control and item bank utilization. Once the system shapes the module for the next stage, it is administered to the examinee, and the whole process is repeated until the last stage is administered.

The new MST method, hereafter referred to as *MST by shaping* (MST-S), combines the unique properties of both CAT and traditional MST designs, which will be referred to as *MST by routing* (MST-R) to distinguish it from the new MST-S. A series of simulation studies were conducted to evaluate the performance of MST-S, comparing it both to MST-R and typical CAT conditions.

Figure 3. Illustration of Multistage Testing by Shaping



Methodology

Simulation Design

The MST-R condition served as a baseline for simulation design, using the 1-3-3 design. Each module contained 20 items. The first-stage routing module consisted of items with a wider range of difficulties; the second and third stages each consisted of three modules with varying levels of difficulty (easy, medium, and hard). A total of 120 items was used to construct the 1-3-3 MST. The items were derived from an item bank of multiple choice items measuring quantitative reasoning skills in an operational CAT program for higher education. Figure 1, introduced as an example earlier in this paper, displays the structure of the stages and information functions for each module. After administration of each stage, individual examinee's interim θ estimate was computed, and a module expected to result in the maximized information function at the estimate was selected for the following stage. For exposure control, two additional panels were constructed consisting of items with identical item characteristics and routing rules (in practice, it would be unrealistic to assume that all panels have items with the exact same characteristics, but this was done for the research purposes in this study). Thus, the MST-R condition included a total 420 items (20 items per module × 7 modules per panel × 3 panels = 420).

Two different CAT conditions were conducted for comparison purposes. The first CAT condition used the maximum Fisher information (CAT-MFI) criterion for item selection. For exposure control, the 'randomesque' method (Kingsbury & Zara, 1989) was used, and one of the best three items based on the MFI criterion was randomly selected and administered (the 1/3 random factor was chosen for its similarity to the MST-R condition, in which one of the three panels was randomly chosen and administered). The second CAT condition used the *a*-stratification method with *b*-blocking (CAT-*a*Str; Chang & Ying, 1999; Chang, Qian, & Ying, 2001). Although the stratification method is already designed to control item exposure by stratifying the item bank, the randomesque method was applied as an additional exposure control method in this condition as well (randomly selecting one of three best items). The item bank was stratified into three item strata and included the same 420 items used to create the MST-R condition.[1]

For MST-S, the main focus of this study, the test consisted of three stages with 20 items each, the same as the MST-R condition. Target TIFs were established for each stage and were set at three evaluation points on the θ scale: θ - 1, θ, and θ + 1. For the first stage, the TIF targets were 4, 5, and 4. For the second and third stages, target TIFs were 9, 15, and 9, and 12, 25, and 12, respectively. These targets were established based on the cumulative TIFs of modules set for the MST-R (Figure 2) condition, allowing the MST-S condition to aim for a comparable level of measurement precision (the peak target TIF value of 25 at θ for the third stage is translated into 0.20 of the standard error of θ estimation). For the module-shaping process of MST-S, three different conditions were studied: 3, 6, and 100 iterations, which, hereafter, will be referred to as MST-S3, MST-S6, and MST-S100, respectively.

Data

Sixty thousand simulees were randomly drawn from a uniform distribution ranging from -3 and 3. The same set of simulees was used in all six studied conditions (MST-R, CAT-MFI, CAT-*a*Str, MST-S3, MST-S6, and MST-S100). The initial θ value for selecting the first item (for CAT-MFI, CAT-*a*Str, and MST-R) was a random number drawn from a uniform distribution ranging from -0.5 and 0.5. During simulating test administrations, interim θ estimates ($\hat{\theta}$) were computed using the expected a posteriori (EAP) estimation method. Once the final stage finished, final θ estimates were computed using the maximum likelihood estimation (MLE) method.

Evaluation

Upon completion of all simulations, conditional standard errors of estimation (CSEE), conditional mean absolute error (CMAE), and conditional bias statistics were computed to evaluate the measurement performance of the studied methods. Those

[1] Because the item pool consisted of items collected from the preassembled MST-R item modules, it was not necessarily optimized for the CAT studies' conditions.

conditional statistics were conditionalized on θ levels, and the width of the θ interval was 0.1. The level of item exposure / pool utilization was also evaluated.

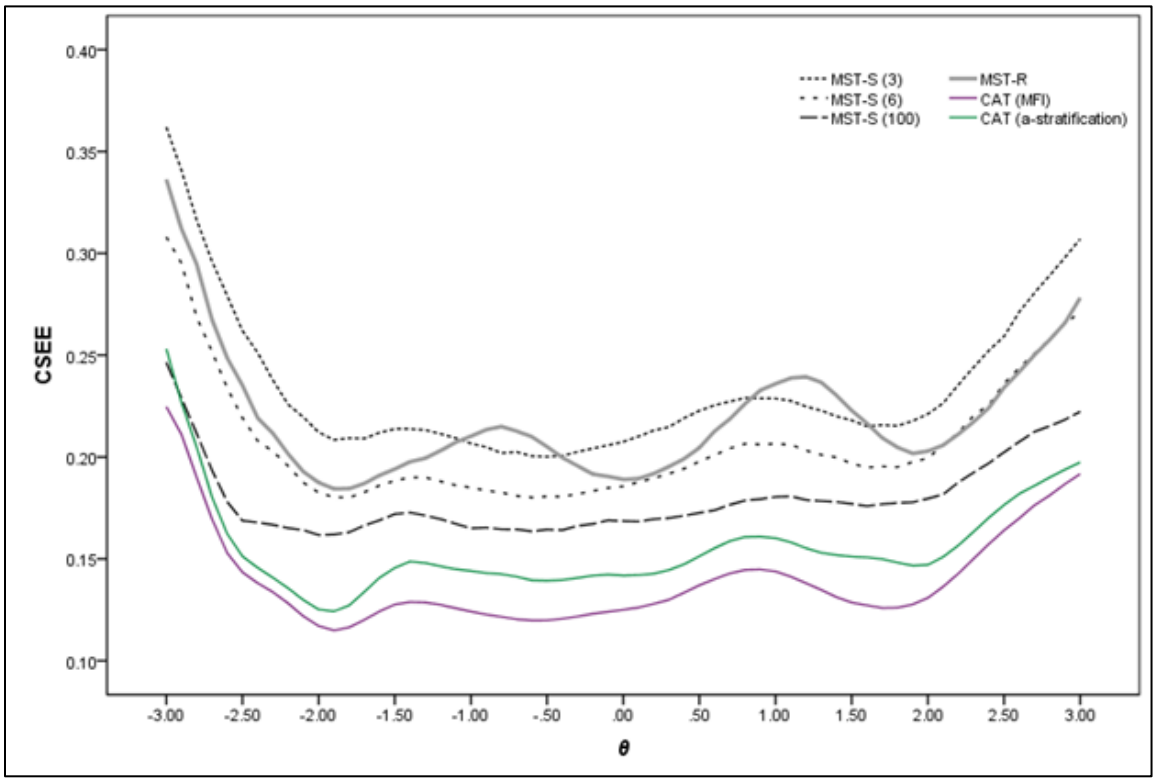
Results

Measurement Performance

Figure 4 displays the CSEE, which essentially is the inverse of the square root of final TIF across θ. Under the MST-R condition (solid gray curve), noticeable bumps in CSEE were observed between -1.5 and -0.5 and between 0.5 and 1.5. Given the MST-R design and modules shown in Figure 2, the fluctuations in CSEE with the MST-R did not differ much from expected results.

The CAT-MFI condition showed a CSEE that was much lower than the MST-R condition, which, again, was no surprise given the fact that the MFI method always looks for items that maximize the information function. The CAT-*a*Str condition resulted in a similar CSEE pattern, but the overall CSEE level was slightly higher than that seen in the CAT-MFI condition.

Figure 4. Conditional Standard Errors of Estimation for Final θ Estimation

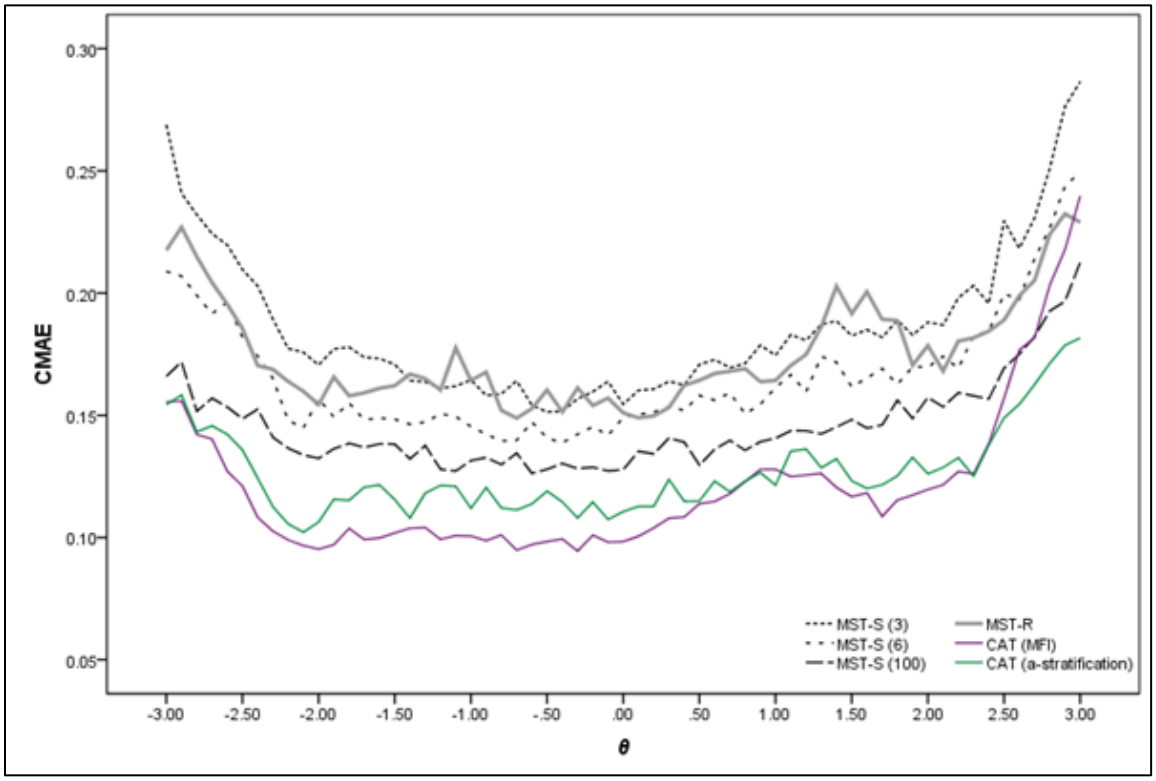


Three different MST-S conditions were studied, each differing in the number of shaping iterations. When the module-shaping process was set to iterate three times (MST-S3), the overall CSEE was comparable to the MST-R condition but slightly higher in many θ areas. With the six iterations for shaping (MST-S6), the CSEE was lower than the MST-R condition for most θ areas. When the shaping iteration was increased to 100 (MST-S100), the resulting CSEE was between the MST-S6 and CAT-*a*Str conditions. Looking at the MST-S3, MST-S6, and MST-100 conditions, it was apparent that the more iterations for module-shaping process, the lower the CSEE. It should be noted that the increase in the number of shaping iterations does not necessarily lower the overall CSEEs; with the increased shaping iterations, the shape of the module is more likely to be closer to the TIF target. If the TIF target was lower, the overall CSEEs would have been increased toward the target as the number of shaping iterations increased. It is also worth noting that the CSEE observed on the MST-S conditions was much flatter (fewer fluctuations) throughout the observed θ areas than that seen under the MST-R and CAT conditions (especially with more iterations for

shaping). This result would seem to indicate that the new MST-S approach is effective in controlling the final TIF and SEE regardless of an examinee's proficiency level.

The conditional standard error of measurement was evaluated based on the conditional mean absolute error (CMAE). As shown in Figure 5, the overall patterns of the CMAE under each studied condition were similar to the CSEE patterns shown in Figure 4. In terms of estimation bias, all studied conditions showed practically none when $\theta = 0$. Under all studied conditions, θ tended to be underestimated when $\theta > 0$ and overestimated when $\theta < 0$, which essentially would shrink the scale of θ. The absolute magnitudes of the biases in Figure 6, however, were too small to be a major concern in practice (less than ± 0.1 between -2.5 and 2.5 except for the MST-R condition, which showed slightly more biases around $\theta = 1.5$).

Figure 5. Conditional Mean Absolute Error for Final θ Estimation

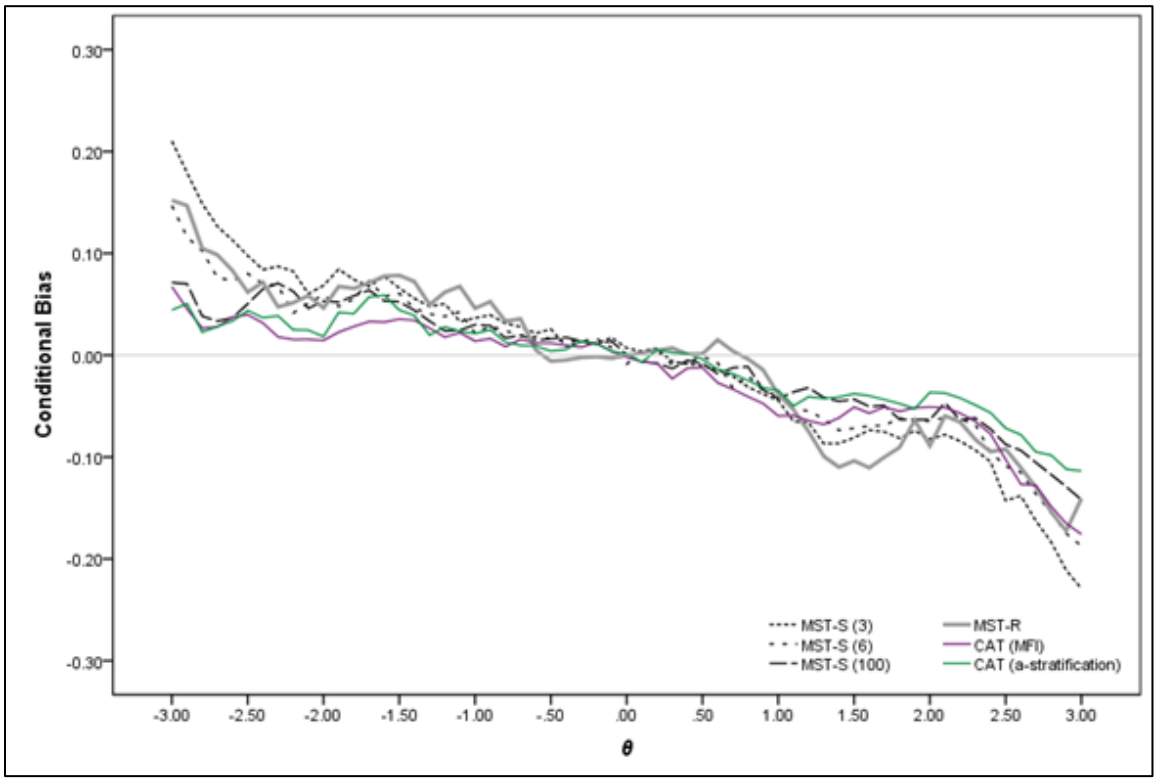


Item Pool Utilization

In Figure 7, the 420 items used in all six studied conditions were ordered by *a*-parameter values (the smallest on the left and the largest on the right of the x-axis) and plotted with exposure rates. Under the MST-R condition, 60 items in the routing module for the first stage (20 items per panel × 3 panels) showed the exposure rate of 0.33, which was exactly as expected under the MST-R design with three panels. All other items used in the modules for the second and third stages did not exceed the exposure rate of 0.13. The maximum observed exposure rate (0.33) of the MST-R condition therefore served as a baseline (the dotted horizontal lines).

Under the CAT-MFI condition, the exposure pattern exhibited the same tendency of the MFI method, favoring items with higher *a*-parameter values. Seventeen items exceeded 0.33 in the CAT-MFI condition, and the maximum observed exposure rate was 0.80. On the other hand, 96 items (23% of the item pool) with lower *a*-parameter values ended up not being used at all. These results from the CAT-MFI condition concur with existing literature that points out the inefficiency of the MFI method in item pool utilization (Georgiadou, Triantafillow, & Economides, 2007; Stocking, 1993).

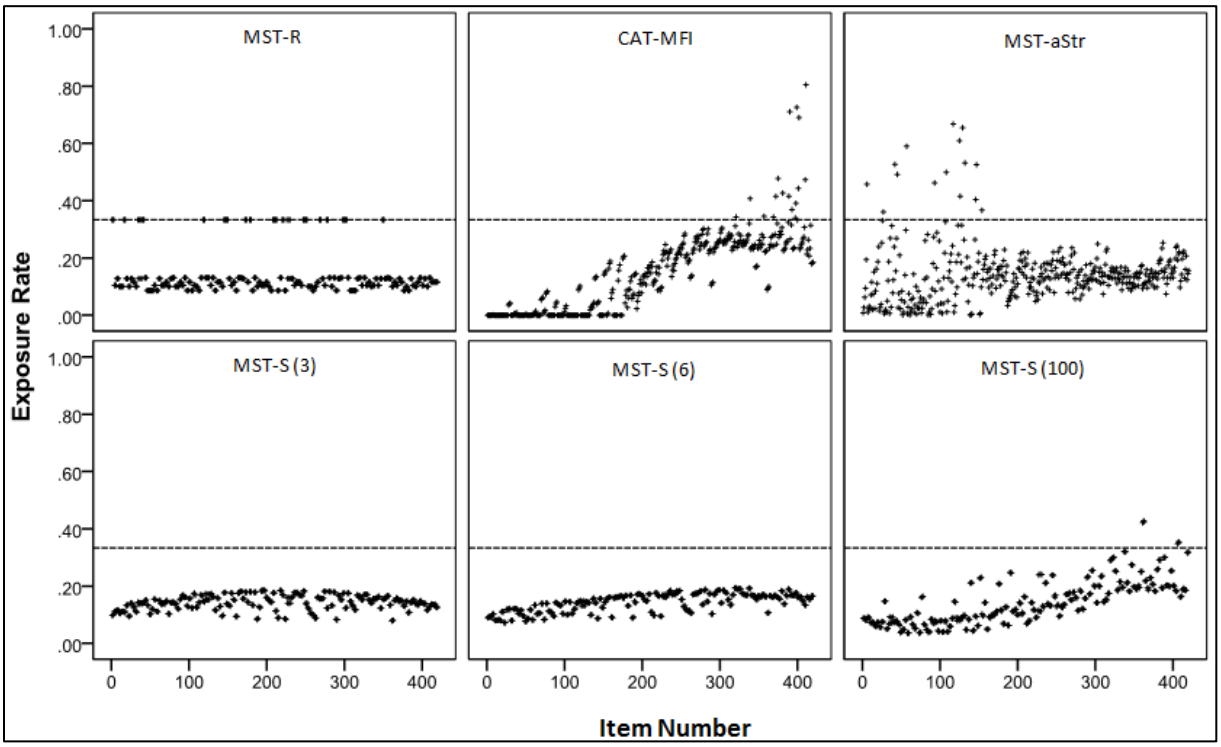
Figure 6. Conditional Bias for Final θ Estimation



With the *a*-stratification method (CAT-*a*Str), there were no unused items; however, several items were used only rarely (Figure 7). Unlike the CAT-MFI condition, the CAT-*a*Str condition showed no tendency to use items with higher *a*-parameter values any more frequently than others. Rather, some items with lower *a*-parameter values showed higher exposure rates, up to 0.67. The main reason for such an item exposure pattern in the CAT-*a*Str condition is that the item pool was stratified into three item strata, and the item stratum used in the beginning was the one with lower *a*-parameter value. At the early stage of CAT, the interim θ estimate and item selection are heavily influenced by the initial θ value, which was a random value between -0.5 and 0.5. Therefore, among the items in the first item stratum, those items whose difficulty was close to zero were used more often than items whose difficulty was far from zero.

Items were used much more evenly with the new MST-S method than they were with the other CAT and MST-R conditions. Under the MST-S3 and MST-S6 conditions, there were no unused items, and the maximum exposure rate was below 0.19. As shown in Figure 7, the MST-S method seems to use items with higher *a*-parameter values more frequently as the number of shaping iterations increased to 100 (MST-S100). It should be noted, however, that increases in the number of shaping iterations does not necessarily make the MST-S method use more items with higher *a*-parameters. If the target TIF had been lower than the ones set in the studied conditions, the increase in the number of shaping iterations would have caused the MST-S method to use more items with lower *a*-parameters.

Figure 7. Item Exposure Rates (Items Ordered by a-Parameter Value)



Discussion

The main purpose of the simulation study was not to compare the MST-S method to other MST-R and CAT methods and determine which one performs the best but to understand how the new MST-S works in typical testing scenarios. The results of the studied conditions should not be imprudently generalized or taken as typical cases for each method. As mentioned earlier, numerous variations in MST-R and CAT designs are possible (Lord, 1980; Zenisky et al., 2010), and even a small change in exposure control and/or item pool composition, for example, can have a major impact on the outcome. Therefore, it is important to use the results from the studied conditions that served as baselines (MST-R, CAT-MFI, and CAT-*a*Str) only as a means to understand the simulation environment where the MST-S was evaluated.

Based on the overall simulation results, it is apparent that the new MST-S approach offers a feasible solution for MST by shaping modules for each stage on the fly. Under studied conditions, the MST-S method was able to achieve measurement precision comparable to the MST-R condition after only three iterations of the shaping process. With six iterations of the shaping process, the MST-S resulted in CSEE and CSEM that were very close to the target and stable throughout the θ scale of interest. Also, the shaping algorithm repeating a random drawing of items turned out to be remarkably effective not only in controlling item exposure but also in utilizing the whole item pool.

As mentioned earlier, the MST-S approach addresses several issues with traditional MST-R and CAT while retaining unique advantages of both MST-R and CAT. Unlike MST-R, MST-S item modules do not need to be preassembled and the module is shaped on the fly according to the autocentered TIF target, resulting in final TIFs for individuals that are much more consistent regardless of examinees' proficiency level. Again, unlike MST-R, all eligible items can be considered for use at every stage in MST-S, which greatly improves the overall level of item pool utilization. Like MST-R, however, MST-S still administers a group of items for each stage and allows examinees to move back and forth and change their responses within each stage. In most CAT programs, which are typically of fixed test length, the measurement precision (i.e., SEE for final θ estimate in operational definition) is not strictly controlled. Some

CAT programs do control SEE by terminating CAT administration once it reaches a target SEE, but then it often creates other problems related to inconsistency in test time and content specifications. On the other hand, while it adaptively constructs tests on the fly just like CAT, MST-S still can provide effective means of managing measurement precision based on the target TIF even when the test length is fixed. In addition, the module-shaping algorithm for MST-S integrates several CAT components for exposure control and content balancing within a single process, which results in substantial simplification of the overall adaptive algorithm.

Of course, MST-S is not a one-size-fits-all solution. Because MST-S essentially retains the multiple stage structure, it may not be as adaptive as a typical CAT that selects an item after each item administration if the number of stages is too small. If measurement efficiency is the major concern and there is no need for item exposure control (for example, as in a brief self-report evaluation for symptoms in an emergency room), a CAT that uses the best items may be the more suitable choice. For testing programs in which local dependence among test items is the major concern, MST-R with preassembled modules, each of which are thoroughly reviewed by test measurement experts before test administration, could be a more appropriate solution over MST-S. Test developers need to consider carefully what they want to achieve from the test design before choosing the test mode.

In this paper, the iterative module-shaping process with repeated random drawings was used to shape the modules for MST-S. Simulation results suggested that this method effectively addressed both item pool utilization and item exposure control issues while quickly realizing the target TIF (by fitting to a module mold) within a few iterations. This takes only a fraction of a millisecond on typical modern PCs. There are, however, a number of different ways to shape the module for each stage on the fly. Basically, any automated test assembly approach, such as the mixed integer programming or greedy methods in conjunction with additional exposure control components, could be used to shape a test module based on the computed module mold for each stage as long as the process can be done fast enough to be on the fly on typical client computers. This would be an interesting area for future studies.

Contact Information

For questions or comments regarding study findings, methodology or data, please contact the GMAC Research and Development department at research@gmac.com.

The views and opinions expressed in this article are those of the author and do not necessarily reflect those of the Graduate Management Admission Council®.

Acknowledgements

The authors wish to thank Dr. Lawrence M. Rudner, Vice President, Research and Development, Graduate Management Admission Council (GMAC®), for his valuable comments that strengthen the paper, and Paula Bruggeman, Writer/Editor, Manager, Research and Development, GMAC for her editorial review.

References

Breithaupt, K., Ariel, A. A., & Hare, D. R. (2010). Assembling an inventory of multistage adaptive testing systems. In W. J. van der Linden, & C. A. W. Glas (Eds.), *Elements of adaptive testing*. New York: Springer Science+Business Media.

Chang, H.-H., & Ying, Z. (1999). Alpha-stratified multistage computerized adaptive testing. *Applied Psychological Measurement, 23,* 211–222.

Chang, H.-H., Qian, J., & Ying, Z. (2001). Alpha-stratified multistage computerized adaptive testing with beta blocking. *Applied Psychological Measurement, 25*, 333–341.

Georgiadou, E., Triantafillow, E., & Economides, A. (2007). A review of item exposure control strategies for computerized adaptive testing developed from 1983 to 2005. *Journal of Technology, Learning, and Assessment, 5*(8). Retrieved October 19, 2012 from http://ejournals.bc.edu/ojs/index.php/jtla/article/view/1647

Jodoin, M. G., Zenisky, A., & Hambleton, R. K. (2006). Comparison of the psychometric properties of several computer-based test designs for credentialing exams with multiple purposes. *Applied Measurement in Education, 19*(3), 203–220.

Kingsbury, G. G., & Zara, A. R. (1989). Procedures for selecting items for computerized adaptive tests. *Applied Measurement in Education*, *2*(4), 359–375.

Lord, F. M. (1980). *Applications of item response theory to practical testing problems.* Mahwah, NJ: Lawrence Erlbaum Associates.

Luecht, R. Brumfield, T., & Breithaupt, K. (2006). A testlet assembly design for adaptive multistage tests. *Applied Measurement in Education, 19*(3), 189–202.

Luecht, R. M., & Nungester, R. (1998). Some practical examples of computer-adaptive sequential testing. *Journal of Educational Measurement, 35*, 239–249.

Melican, G. J., Breithaupt, K., & Zhang, Y. (2010). Designing and implementing a multistage adaptive test: The Uniform CPA exam. In W. J. van der Linden, & C. A. W. Glas (Eds.), *Elements of adaptive testing*. New York: Springer Science+Business Media.

Stocking, M. L. (1993). *Controlling item exposure rates in a realistic adaptive testing paradigm*. Technical Report RR 3–2. Princeton, NJ: Educational Testing Service.

Stocking, M. L., & Lewis, C. (1995). A new method of controlling item exposure in computerized adaptive testing. *Research Report* 95–25. Princeton, NJ: Educational Testing Service.

Stocking, M. L., & Lewis, C. (1998). Controlling item exposure conditional on ability in computerized adaptive testing. *Journal of Educational and Behavioral Statistics, 23*(1), 57–75.

Sympson, J. B. & Hetter, R. D. (1985). Controlling item-exposure rates in computerized adaptive testing. In Proceedings of the 27th annual meeting of the Military Testing Association, (pp. 973–977), San Diego, CA: Navy Personnel Research and Development Centre.

van der Linden, W.J. (2005). *Linear models for optimal test design*. New York: Springer.

Weiss, D. J. (1974). Strategies of adaptive ability measurement. (*Research Report 74*-5). Minneapolis: University of Minnesota, Department of Psychology, Psychometric Methods Program.

Zenisky, A., Hambleton, R. K., & Luecht, R. M. (2010). Multistage testing: Issues, designs, and research. In W. J. van der Linden, & C. A. W. Glas (Eds.), *Elements of adaptive testing*. New York: Springer Science+Business Media.