A Hybrid Strategy to Construct Multistage Adaptive Tests

Applied Psychological Measurement
2018, Vol. 42(8) 630–643
© The Author(s) 2018
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0146621618762739
journals.sagepub.com/home/apm

Xinhui Xiong¹

Abstract

How to effectively construct multistage adaptive test (MST) panels is a topic that has spurred recent advances. The most commonly used approaches for MST assembly use one of two strategies: bottom-up and top-down. The bottom-up approach splits the whole test into several modules, and each module is built first, then all modules are compiled to obtain the whole test, while the top-down approach follows the opposite direction. Both methods have their pros and cons, and sometimes neither is convenient for practitioners. This study provides an innovative hybrid strategy to build optimal MST panels efficiently most of the time. Empirical data and results by using this strategy will be provided.

Keywords

multistage adaptive tests (MSTs), panel, bottom-up method, top-down method, hybrid method

Introduction

A multistage adaptive test (MST) includes multiple modules that each of them is administered at different stages during the testing administration based on examinees' prior performance. A module (Luecht, 1998; Luecht & Nungester, 1998) is a set of items administered as a test unit. Although a traditional computerized adaptive test (CAT) adaptively selects individual items, a MST adaptively selects individual modules at each stage. A MST usually starts with an initial module at the first stage for all examinees. Based on the performance of the initial module, examinees are routed to different modules at the second stage. Those modules share the same/similar content distribution but vary at the difficulty levels. The from-the-current-to-the-next-stage process repeats until examinees reach the module of the last stage. All sets of modules, from the module of the beginning stage to the modules of the last stage, constitute a panel for a MST, which includes all possible item sets that an examinee could have taken. The reasons that MSTs have seen increased adoption with testing companies during the past 20 years come primarily from two sources. First, conventional linear tests are usually composed of items appropriate for typical examinees, and are lengthy to ensure sufficient ability estimates accuracy for atypical examinees. Also, conventional linear tests have fixed test length and are preassembled as a single form or multiple parallel forms, which means all examinees are administered the

¹American Institute of Certified Public Accountants, Ewing, NJ, USA

Corresponding Author:

Xinhui Xiong, American Institute of Certified Public Accountants, Princeton South Corporate Center, 100 Princeton South, Suite 200, Ewing, NJ 08628, USA.
Email: xinhui.xiong@aicpa-cima.com

same series of items. On the contrary, item-level adaptive tests select each item based on the individual's performance on the prior items (van der Linden & Cees, 2000; Weiss, 1974). Therefore, item-level adaptive tests are normally shorter with the same ability estimate accuracy (Vispoel, Rocklin, & Wang, 1994; Vispoel, Wang, & Bleiler, 1997), yet at a price of much more complicated assembly and delivery algorithms. As a result, MSTs, a compromise between conventional linear tests and item-level adaptive test, have gained popularity these days: They are shorter than conventional linear tests in general, and with much less cost compared with CATs (Cronbach & Gleser, 1965; Wainer & Kiely, 1987, Lewis, Lee, & von Davier, 2012).

Although the topic of optimal strategies to construct conventional linear forms and item-level adaptive test forms has been well researched (Ackerman, 1989; Armstrong, Belov, & Weissman, 2005; Belov & Armstrong, 2006; Boekkooi-Timminga, 1990; van der Linden, 2010; van der Linden & Reese, 1998), few studies have been conducted focusing on methods for constructing optimal MSTs panels. This study proposes a new hybrid strategy to construct MSTs. The strategy has been applied to a large-scale international licensure examination, and the resulting panels are completely acceptable for practical use. This strategy can be easily applied to any other MSTs beyond certification context.

Test Background

The examination includes four separate sections (denoted as Section A, Section B, Section C, and Section D). All four sections are administered quarterly. This means that test assembly occurs four times per year. For each test window, when test assembly is completed, a test review process is started to ensure the final quality of the test forms that will be delivered to candidates. Two test structures exist: An old test structure that had been used since 2011, and a new test structure that started in April 2017.

Test Structure

This research was conducted on the old test; therefore, only the old test structure will be introduced. For the old test structure, Sections A, B, and C are composed of two types of items: multiple-choice questions (MCQs) and task-based simulations (TBSs). A TBS is defined as "the assessment of knowledge and skills in a context approximating that found on the job through the use of realistic scenarios and tasks, and access to normally available and familiar resources" (DeVore, 2002, p. 2). Each TBS includes a group of measurement opportunities (MOs). A MO is an aspect of a TBS that is designed to be scored, and examinees respond to each MO as part of a TBS. Section D is composed of constructed response (CR) items and MCQs. Usually 24 panels are preassembled for each section per test window. For Sections A, B, and C, each panel includes five MCQ modules and one TBS module (Figure 1). For Section D, each panel includes five MCQ modules and one CR module (Figure 2). The first three stages are for MCQ items; the fourth stage is for TBS items for Sections A, B, and C, and for CR items for Section D. Hence, there are four stages in total. For the five MCQ modules in one panel, three of them are of moderate difficulty level (M1, M2, and M3 in Figures 1 and 2), and two of them are at hard difficulty level (D2 and D3 in Figures 1 and 2). An examinee starts with an initial module at the moderate difficulty level (M1 in Figures 1 and 2) at Stage 1; based on his or her performance at the first stage, the examinee is routed to the second stage with either another moderate module (M2 in Figures 1 and 2) or a hard module (D2 in Figures 1 and 2). After the second stage, the examinee is routed to the third stage with either a moderate module (M3 in Figures 1 and 2) or a hard module (D3 in Figures 1 and 2) at the third stage based on his or her performance of the prior two modules. After that, examinees are routed to a TBS

Figure 1. Old test structure for Sections A, B, and C.
Note. M and D are for OP items. MCQ = multiple-choice questions; TBS = task-based simulation; M = medium; D = difficult; PT = pretest; OP = operational.

Figure 2. Old test structure for Section D.
Note. M and D are for OP items. MCQ = multiple-choice question; CR = constructed response; M = medium; D = difficult; PT = pretest; OP = operational.

module for Sections A, B, and C, and to a CR module for Section D at Stage 4. It is obvious that there are four routes in each panel: MMM, MMD, MDM, and MDD. MMM refers to a route with a moderate module at each stage. MMD refers to a route with moderate modules for the first two stages and a difficult module at the third stage. A route is a possible path in one panel from the initial stage to the last. Each examinee is assigned one route in one panel. Note that only MCQ part is adaptive for this test. A test form includes three MCQ modules, plus one TBS module for Sections A, B, and C, and one CR module for Section D.

Method

A test specification is a document providing all requirements for a test, including all statistical requirements and all content requirements. A blueprint defines all content knowledge and skills to be tested. Hence, a blueprint can be served as a starting point of a test specification. From this starting point, statistical requirements are added to define a complete test specification. Here are

some examples of content requirements: how long the whole test is, how many item types are required, the number of items for each item type. Examples of statistical requirements include the average difficult level of a whole test, the distribution of test item difficulty level, and so on. Collecting test specification should be done before the test construction.

Presently, the most commonly used approaches for MST form construction fall in two categories: bottom-up and top-down (Luecht, 1998; Luecht & Nungester, 1998). The bottom-up approach splits the whole MST into several modules. The first step is to construct each module, and then compile those modules to obtain the whole test. With this method, test specifications for each module should be provided. For the old test structure, test specifications for modules at moderate difficulty level (Model M1, M2, and M3) and modules at hard difficulty level (Module D2 and D3) should be provided. Based on specifications at module level, a group of modules at moderate and hard difficulty levels is assembled first. Note that modules at the same difficulty level are exchangeable in this case. Then, the modules assembled in the first step are mixed and matched to build panels. This approach is similar to human perception forming process: eyes catch the bottom information, and the brain gets the whole image to be recognized as perception. Because of this resemblance, this approach is widely used by researchers. Alternatively, the top-down approach formulates an overview of the whole test, starts with the whole picture of the test, then breaks down to module level. With this method, test specification for the whole test, not for each module, is required. And, modules have to be combined in certain ways to reach goals at test level. In this case, modules at the same difficulty level are not necessary exchangeable. Therefore, top-down method makes the assembly for MSTs especially complicated: A panel needs to be built first, and then divide the panel into certain number of modules that meet predefined rules. Our old test used bottom-up method. For the old test structure with bottom-up method, M1, M2, and M3 have to be assembled in a way that the three pieces together meet desired features. So do MDD, MDM, and MMD.

Although the bottom-up approach is reasonably intuitive, the challenge is how to divide “specifications” at test level into the corresponding modules (Batagelj, Doreian, & Ferligoj, 1992; Doreian & Mrvar, 1994; Luecht, 1998; Luecht & Nungester, 1998). For example, with the old test, senior technical managers (STMs) only provide the required number of MCQ pretest (PT) items for each content area for one test window (each item belongs to one content area). That is, the number is counted based on all assembled panels together. In this case, the information is provided neither at the panel level nor at the module level. Therefore, when MST forms for each section each test window are assembled, the exact number of PT items in each content area for each panel and for each MCQ module are unknown. And that makes the first step of the bottom-up method, module assembly, not easy to be accomplished. In the same vein has to do with the specifications on the percentage of items from each content area in each route. The blueprint only provides information on the percentages of each content area on one route. Assemblers have no idea of the percentages at the module level. By using bottom-up strategy, one way to obtain those numbers at module level is to calculate them manually through other numbers known. Let us go back to the first example. The range of the number of PT MCQ items in each content area for one module can be calculated based on the number of panels to be built, the number of moderate modules to be built, the number of difficult modules to be built, the number of PT MCQ items in each area in total, and the reuse times for each individual MCQ PT item (the number of panels that can use the same item in one test build). Note that this procedure does not always lead to a single number at the module level, but a range, which is the biggest shortcoming of this approach. That means the exact number STMs provided cannot always be guaranteed, and it usually ends up with a number close to the original number requested by a STM. Also, as you can tell, this calculation procedure does not fit all situations, which can be illustrated by the following example: The percentage of one area in one route includes MCQ

items and MOs of TBS items (CR items do not count in this scenario). No simple calculation can be used to obtain percentages at the module level (include multiple MCQ modules and one TBS module).

In contrast, content balance and statistical properties for each module might be lost with the top-down strategy. As mentioned before, after a panel with all nice features meet at panel level is built, how to divide this large item set into different modules is challenging. For example, when the panel provides desired test information function (TIF) curves for each route, it is challenging to collapse the whole item set into multiple modules with each module at same difficulty level keeps similar location on module TIF. So, it requires significant planning to properly set constraints at test levels across stages (Adema, 1990). Another example is, if the blueprint requires a test include 25 (50%) items from Standard 1, 15 (30%) items from Standard 2, and 10 (20%) items from Standard 3, with the top-down strategy, there is no guarantee to the standard balance in each module for each stage unless extra actions are taken.

In contrast to the bottom-up and top-down approaches, the hybrid strategy can produce acceptable results without significant drawbacks. The core idea of this strategy is to classify requirements on test forms into different levels, such as item level, item-set level, route level, and so on. Under this strategy, test forms are not initially assembled at module level, nor assembled at the test level first. The assembly process deals with test requirements at different levels simultaneously. More specifically, six levels can be get involved: solution level, panel level, route level, module level, item-set level, and item level. A solution is the output of a single run from the test-form-generation system. It usually includes a number of parallel panels. For the old test, 24 panels are usually built simultaneously. In this case, 24 panels is one solution. Every requirement for the test is attached to one level. Not all tests include requirements at all six levels. Example requirements at each level are provided below.

Sample requirements at *solution* level:

- the number of panels;
- the number of MCQ PT items for each area;
- the number of TBS PT items;
- the minimal and maximal reuse times for each MCQ/TBS operational/PT items across panels;
- the number of MCQ/TBS operational items that have been used within certain number of prior test windows due to test security concern;

Sample requirements at *panel* level:

- the number of stages;
- the number of moderate/hard MCQ modules;
- the number of TBS modules;
- the number of CR modules.

Sample requirements at *route* level include (module difficulty level and stage number are combined together to refer to specific modules. For example, M1 refers to moderate modules at Stage 1.):

- no overlap items between modules in the same route and the same panel. For example, M1, M2, and M3 in one panel should not include duplicate items;

- a couple of overlap items are allowed between modules in the same panel and the same stage but in different routes. For example, a few items shared by M3 and D3 in one panel are acceptable;
- item percentage for each standard level;
- route information curves should be within an acceptable range of the target and be as close as possible to each other.

Sample requirements at *module* level include:

- the number of operational/PT items in each MCQ/TBS/CR module;
- no enemy items within one module. Enemy items are items that cover the same material or have options to cue answers of other questions. The set of enemy items are coded with the same enemy code. This requires that each module can include at most one item with the same enemy code under the same standard and substandard;
- at most one item with the same standard and substandard can be selected within one module for TBS module;
- the location and shape of the module information.
- *Note.* Module information is the Fisher's information for one module.

Sample requirements at *item-set* level include:

- a group of items that need to be selected into one solution together;
- a group of items that need to be selected and put into one module together.

Finally, sample requirements at *item* level include:

- over exposed items are filtered;
- items with lower exposure should be selected when other item properties are comparable.

Software

For the old production system, IBM ILOG CPLEX Optimization Studio 12.6.1 was used as optimization engine. This research used the same CPLEX Optimization Studio. Mixed integer programming (MIP) technique (van der Linden, 2005) was used for both the old production system and the research.

Mathematic Models

Constraints used in the research included all the requirements from the test specification and each constraint attached to one level. The conceptual objective was set to maximize Fisher's information within certain ranges for the two main routes (MMM and MDD) of each panel in the resulting solution. In the following text, this Fisher's information is referred to as the actual route information, and the certain ranges are the target route information ranges. A target route information range include a target minimal and maximal Fisher's information for the route. Five predefined locations for each of the two main routes were used to control the maximization. The actual route information was calculated for the three MCQ modules in the route. This conceptual objective was implemented in CPLEX model in this way as follows: minimize the deviation of the actual route information from the target route information range for each route in each

panel and a uniform minimal deviance (y) is used for both routes. The primary reason that two out of four routes were focused on is that a majority of the test candidates take either MMM or MDD. The other reason is to decrease the system complexity. The predefined locations and the target route information range corresponding to each location are set based on historical data. So, there are five target theta points for MMM and MDD routes separately, and five target information ranges corresponding to those target theta points. Therefore, for each theta point, a target range of Fisher's information was provided. For example, $\theta_{m1}, \theta_{m2}, \theta_{m3}, \theta_{m4}$, and θ_{m5} are five target theta points for MMM route, and $\theta_{d1}, \theta_{d2}, \theta_{d3}, \theta_{d4}$, and θ_{d5} are five target theta points for MDD route. For each target theta point, a target minimal and maximal Fisher's information for the route are specified. Mathematical formula for controlling the actual route information for each moderate route in each panel is provided below:

Minimize y ;

Subject to:

$$\sum_{i=1}^I I_i(\theta_k) x_{pi} \leq TL_k - W_k \times y, \text{ for each MMM route, each } k \text{ in } \{m1, m2, m3, m4, m5\}$$

$$\sum_{i=1}^I I_i(\theta_k) x_{pi} \geq TL_k - W_k \times y, \text{ for each MDD route, each } k \text{ in } \{d1, d2, d3, d4, d5\}$$

I is the number of MCQ items in one route. y is the gap between target information and information of the solution. The same y applied to all panels in one solution. x_{pi} are decision variables: If item i was selected by a panel p , $x_{pi} = 1$; otherwise, $x_{pi} = 0$. TU_k is the upper bound of the targeted information at θ_k , and TL_k is the lower bound of the targeted information at θ_k . $I_i(\theta_k)$ is the item information at θ_k . W_k is the weight set for each target theta point. The weights offer the freedom for the test assembler so more focus can be put on the more important theta area(s), such as area that is closer to the cut. Similar formula for controlling Fisher's information for difficult routes was used. Below are some other examples from the research model.

For each item i ,

$$\sum_{p=1}^P x_{pi} \leq PU,$$

$$\sum_{p=1}^P x_{pi} \geq PL.$$

For each selected item i , it should be used by at least PL panels in one solution, and by no more than PU panels in one solution. For operational items, this constraint is used to control the exposure rate. For PT items, this constraint is used to ensure enough exposure to obtain sample size large enough for item parameter calibration.

For each module in one panel (p),

$$\sum_{i \in v_{mo}} x_{pi} = n_{mo},$$

$v_{mo} = \{\text{operational items in one MCQ module}\}$, n_{mo} is the number of MCQ operational items required in one MCQ module;

$$\sum_{i \in v_{mp}} x_{pi} = n_{mp},$$

$v_{mp} = \{\text{pretest items in one MCQ module}\}$, n_{mp} is the number of MCQ PT items required in one MCQ module;

$$\sum_{i \in v_{to}} x_{pi} = n_{to},$$

$v_{to} = \{\text{operational items in one TBS module}\}$, n_{to} is the number of TBS operational items required in one TBS module;

$$\sum_{i \in v_{tp}} x_{pi} = n_{tp},$$

$v_{tp} = \{\text{pretest items in one TBS module}\}$, n_{tp} is the number of TBS PT items required in one TBS module;

$$\sum_{i \in v_{to} \cap i \in v_{sp}} x_{pi} = 0,$$

$v_{sp} = \{\text{certain item type}\}$. This constraint is to prevent certain item types from selection for TBS operational items in one TBS module;

$$\sum_{i \in v_{all} \cap i \in v_{spe}} x_{pi} = 0,$$

$v_{all} = \{\text{all items in one module}\}$, $v_{spe} = \{\text{one enemy item set}\}$. This constraint is to prevent more than one enemy item of a enemy item set from selection in one module;

$$\sum_{i \in v_{to}} b_i \frac{x_{pi}}{n_{to}} \geq BL_{to},$$

$v_{to} = \{\text{operational items in one TBS module}\}$, BL_{to} is the lower bound of TBS module average difficulty level, b_i is the item difficulty parameter;

$$\sum_{i \in v_{to}} b_i \frac{x_{pi}}{n_{to}} \leq BU_{to},$$

$v_{to} = \{\text{operational items one TBS module}\}$, BU_{to} is the upper bound of TBS module average difficulty level, b_i is the item difficulty parameter;

$$\sum_{i \in v_{tmo}} m_i x_{pi} \geq ML_{tmo},$$

m_i is the MO count for TBS item i . $v_{tmo} = \{\text{operational items one TBS module}\}$, ML_{tmo} is the lower bound of total MO counts in one TBS module.

$$\sum_{i \in v_{tmo}} m_i x_{pi} \leq MU_{tmo},$$

m_i is the MO count for TBS item i . $v_{tmo} = \{\text{operational items one TBS module}\}$, MU_{tmo} is the upper bound of total MO counts in one TBS module.

For each solution,

$$\sum_{i \in v_e} s_i \leq U_e,$$

$$\sum_{i \in v_e} s_i \geq L_e.$$

$v_e = \{\text{items in one solution that belongs to one category}\}$, This constraint restricts the number of items from certain category in one solution. s_i are decision variables: If item i was selected,

Table 1. Item Frequency for Each Item Type in Item Pool (Each Section Has One Item Pool).

Section	OP			PT			Total
	CR	TBS	MCQ	CR	TBS	MCQ	
A		121	2,422		17	497	3,057
B		117	2,916		19	364	3,416
C		85	2,041		15	1,195	3,336
D	93	NA	2,089	11	NA	323	2,516

Note. Only Section D has CR items. And, for the old test, Section D does not include TBS items. OP = operational; PT = pretest; CR = constructed response; TBS = task-based simulation; MCQ = multiple-choice questions.

$s_i = 1$; otherwise, $s_i = 0$. L_e and U_e are the lower and upper bound for the number of items from that category in one solution. For example, this category can be each content area, each content skill level, items that had been used from previous test administration, and so on.

More mathematical formula related to linear models for automated test assembly can be found in van der Linden’s (2005) book.

Research Results

The model was built with the hybrid strategy discussed in the prior section for the old test structure. The primary goal of this research is to adopt the new hybrid strategy and compare the result with the practical result obtained from the old production system with the bottom-up method. Data extracted from operational item pools were used to generate 24 parallel panels for each section.

Table 1 indicates the number of items for each item type in each item pools for each section used for the research. Note that these pools were the exact pools used for practical assembly in the old production system for the old tests using the bottom-up method.

Not surprisingly, the information curves for MMM and MDD routes of 24 panels in one solution generated by the research model fall into the targeted range and stick to each other very well (Figure 3) compared with the solution generated by the old production system (Figure 4). Of course, all other content requirements and statistical requirements were met. Figures 3 and 4 are for Section A. Note that Figure 4 showed the route TIF curves of the solution generated from the old production system. This solution is not the final test delivered to examinees. As mentioned in “Test Background” section, a test review process comes after test assembly to ensure the quality of the final test. Better test assembly leads to less work during the test review process. Descriptive statistics for TIF height gaps (difference between the highest and lowest point at each θ point) between Figures 3 and 4 in Table 2 shows that mean height difference across θ for moderate routes and difficult routes were much smaller in Figure 3 (hybrid methods) than the mean difference in Figure 4 (bottom-up method). Table 3 shows the detailed TIF heights information, maximum, minimum, and difference at each θ point for certain θ range that we care: -1.8 to -0.6 for moderate routes, and -0.9 to 0.3 for difficult route. It is clear that differences were uniformly much lower for results generated from the hybrid method than that from the bottom-up method. The same trend was identified for the other three sections.

A bonus from using the hybrid strategy, compared with the bottom-up method used by the old system, is item pool utilization. Besides the shortcomings discussed earlier in this study, the bottom-up method splits assembly procedure into two steps: first, assemble modules, and then assemble panels based on modules assembled in the first stage. In the past, with the bottom-up

Table 2. Descriptive Statistics of TIF Heights for Figures 3 and 4.

	Bottom-up method	Hybrid strategy
Moderate route		
M	1.08	0.22
SD	0.29	0.17
Maximum	1.49	0.03
Minimum	0.71	0.55
Range	0.78	0.52
Difficult route		
M	0.89	0.26
SD	0.18	0.15
Maximum	1.13	0.00
Minimum	0.67	0.46
Range	0.45	0.46

Note. TIF = test information function.

Table 3. A Comparison of TIF Heights Between Figures 3 and 4 at Focused θ Range.

	Theta	Bottom-up method			Hybrid strategy		
		Maximum	Minimum	Difference	Maximum	Minimum	Difference
Moderate route	-1.8	9.95	8.67	1.28	9.13	9.57	0.43
	-1.7	10.20	8.92	1.28	9.42	9.86	0.44
	-1.6	10.38	9.17	1.21	9.64	10.08	0.44
	-1.5	10.48	9.40	1.08	9.81	10.23	0.42
	-1.4	10.50	9.64	0.86	9.91	10.28	0.37
	-1.3	10.44	9.71	0.73	9.96	10.25	0.29
	-1.2	10.43	9.71	0.72	9.96	10.15	0.18
	-1.1	10.36	9.65	0.71	9.92	10.05	0.13
	-1	10.31	9.52	0.79	9.83	9.94	0.11
	-0.9	10.30	9.21	1.09	9.69	9.77	0.08
	-0.8	10.18	8.86	1.31	9.48	9.54	0.06
	-0.7	9.94	8.48	1.45	9.20	9.25	0.04
Difficult route	-0.6	9.57	8.08	1.49	8.86	8.89	0.03
	-0.9	9.16	8.04	1.13	8.73	8.73	0.00
	-0.8	9.36	8.26	1.11	8.96	8.96	0.00
	-0.7	9.57	8.46	1.11	9.14	9.15	0.02
	-0.6	9.72	8.65	1.08	9.26	9.30	0.04
	-0.5	9.82	8.82	1.00	9.34	9.38	0.05
	-0.4	9.86	8.96	0.90	9.36	9.45	0.09
	-0.3	9.85	9.04	0.81	9.33	9.50	0.17
	-0.2	9.79	9.04	0.76	9.28	9.52	0.25
	-0.1	9.70	9.00	0.69	9.19	9.51	0.31
	0	9.57	8.90	0.67	9.09	9.45	0.36
	0.1	9.45	8.77	0.68	8.98	9.37	0.39
	0.2	9.39	8.64	0.75	8.85	9.26	0.40
	0.3	9.33	8.49	0.85	8.72	9.11	0.39

Note. TIF = test information function.

method, items selected within the last three test windows have a much larger probability of being selected again. That means, items selected before the last three test windows have a slim chance of being used again. In one extreme case, an item was never used as an operational item

Figure 3. Route TIFs for the 24 panels generated from this research model using the hybrid method for the old test (Section A).
Note. TIF = test information function.

Figure 4. Route TIFs for the 24 panels generated from the old production system using the bottom-up method for the old test (Section A).
Note. TIF = test information function.

after being pretested. Items that are used within a specified range of time are active items. The range of time is defined by the people who have oversight over the evaluation of the item pool. If items are not considered active, then they are deemed to be inactive items. A yearly period

Table 4. Percentages of Selected Inactive Items for Each Item Type Each Time Period for Section A Using the Hybrid Method.

The hybrid method				
	1 year	2 years	3 years	4 years
MCQ OP	22.18%	18.06%	13.05%	8.54%
TBS OP	28.57%	18.37%	14.29%	2.04%

Note. Each cell indicates the percentage of inactive items selected from the item pool for each time period. As an example of choosing a 1-year time period for differentiating between active and inactive items, in Section A, 22.18% of selected operational MCQ items were used 1 year (four test windows) earlier and are, therefore, considered to be inactive. MCQ = multiple-choice questions; OP = operational; TBS = task-based simulation.

Table 5. Percentages of Selected Inactive Items for Each Item Type Each Time Period for Section A Using the Bottom-Up Method.

The bottom-up method				
	1 year	2 years	3 years	4 years
MCQ OP	1%	0.17%	0.17%	0.00%
TBS OP	25%	4.17%	0.00%	0.00%

Note. Each cell indicates the percentage of inactive items selected from the item pool for each time period. As an example of choosing a 3-year time period for differentiating between active and inactive items, in Section A, 0% of selected operational TBS items were used 3 years (12 test windows) earlier and are, therefore, considered to be inactive. MCQ = multiple-choice questions; OP = operational; TBS = task-based simulation.

represents four consecutive test windows. In looking to differentiate between active and inactive items, the range of time that is considered for this study is between one and four yearly periods. As an example, for one yearly period, items were used in the four prior test windows were classified as active items, and items used before the four prior windows were classified as inactive items. A comparison of the item pool utilization from the results for Section A using two different methods (the hybrid method and the bottom-up method) is shown in Tables 4 and 5. Table 4 shows the result from the new hybrid strategy, and Table 5 is the result generated by the old system. Again, numbers in Table 5 are not from the final test delivered to candidates, but are from the test assembly solution. In both tables, the number in one cell indicates the percentage of inactive items selected from the item pool for one item type and one time period. As an example of choosing a 1-year time period for differentiating between active and inactive items, in Section A, 22.18% (Table 4) of selected operational MCQ items were used 1 year (four test windows) earlier and are, therefore, considered to be inactive.

A comparison of the two tables (Tables 4 and 5) indicates that the opportunity for selecting inactive items has significantly increased with the hybrid method. For example, for Section A, the percentage of inactive operational TBS items selected increased from 4.17% (Table 5, with the bottom-up method) to 18.37% (Table 4, with the hybrid method) when 2 years is chosen as the time period. It increased from 0% to 14.29% when 3 years is chosen as the time period. The same trend was identified for the other three sections. Further research will be conducted after new test data are received.

Discussion

This hybrid strategy provides an innovative way to construct optimal MSTs in an efficient way. For the bottom-up method, as mentioned earlier, there are two stages. This procedure was challenging if the modules assembled in the first stage were not good enough to meet requirements at the test level because modules assembled in the first step are primarily to meet requirements at module level. It was possible that the panels could not be assembled successfully even though the item pool has the inventory adequacy. On the contrary, it is very difficult for top-down method to keep content balance and statistical properties at module level. The proposed hybrid method overcomes those shortcomings and enables test assemblers to simply input all statistical and content constraints at different level that naturally match what are described in blueprint.

Removing the extra step for test assemblers to calculate parameter values for certain constraints is the first and intuitive benefit offered by the hybrid method. For example, breaking down requirements at higher level, such as solution level, to a lower level, such as module level, is not needed any more.

Second benefit is, compared with both the bottom-up and top-down methods, a better solution can be found with the hybrid method in many cases unless the item pool is not good enough to provide a better solution. This is because both the bottom-up and top-down methods assemble panels in two steps, and the assembly in the second step depends on the results of the first assembly step. Comparisons between Figures 3 and 4, and results in Tables 2 and 3 illustrate the improvement made by the hybrid strategy.

More benefits of the hybrid strategy seems promising, such as better item pool utilization. This can be further supported by future exploration when data of the new test are received.

On the contrary, one consideration before using the hybrid strategy with only one assembly step instead of two assembly steps as used in the bottom-up or top-down strategies is the following: The model using the hybrid strategy includes more constraints and variables at one time. Around 7,700,000 constraints and 460,000 variables are used in this research model. The running time is between 15 and 35 min to get an optimal solution with y less than 0.20. With the same computer (1.9GHz), an acceptable solution usually can be obtained below 5 min with the bottom-up method. Meanwhile, more advanced computer is an option to cut down run time.

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.

Reference

- Ackerman, T. (1989, March). *An alternative methodology for creating parallel test forms using the IRT information function*. Paper presented at the annual meeting of the National Council on Measurement in Education, San Francisco, CA.
- Adema, J. J. (1990). *Models and algorithms for the construction of achievement tests*. Enschede, The Netherlands: University of Twente.
- Armstrong, R. D., Belov, D. I., & Weissman, A. (2005). Developing and assembling the Law School Admission Test. *Interfaces*, 35, 140-151.
- Batagelj, V., Doreian, P., & Ferligoj, A. (1992). An optimization approach to regular equivalence. *Social Networks*, 14, 121-135.

- Belov, D. I., & Armstrong, R. D. (2006). A constraint programming approach to extract the maximum number of non-overlapping test forms. *Computational Optimization and Applications*, 33, 319-332.
- Boekkooi-Timminga, E. (1990). The construction of parallel tests from IRT-based item banks. *Journal of Educational Statistics*, 15, 129-145.
- Cronbach, L. J., & Gleser, G. C. (1965). *Psychological tests and personnel decisions* (2nd ed.). Urbana: University of Illinois Press.
- DeVore, R. (2002). *White paper on simulations*. Unpublished paper.
- Doreian, P., & Mrvar, A. (1994, December). *A partitioning approach to structural balance*. Paper presented at the Social Science Information Technology Conference, Amsterdam, The Netherlands.
- Lewis, Y.-H., Lee, C., & von Davier, A. A. (2012, May). *Test security for multistage tests: A quality control perspective*. Paper presented at the Conference on Statistical Detection of Potential Test Fraud, Lawrence, KS.
- Luecht, R. M. (1998). Computer-assisted test assembly using optimization heuristics. *Applied Psychological Measurement*, 22, 224-236.
- Luecht, R. M., & Nungester, R. J. (1998). Some practical examples of computer-adaptive sequential testing. *Journal of Educational Measurement*, 35, 229-249.
- van der Linden, W. J. (2005). *Linear models for optimal test design*. New York, NY: Springer.
- van der Linden, W. J. (2010). Constrained adaptive testing with shadow tests. In W. J. van der Linden & C. A. W. Glas (Eds.), *Elements of adaptive testing* (pp. 31-55). New York, NY: Springer.
- van der Linden, W. J., & Glas, (Eds.). (2000). *Computerized adaptive testing: Theory and practice*. Dordrecht, The Netherlands: Kluwer Academic.
- van der Linden, W. J., & Reese, L. M. (1998). A model for optimal constrained adaptive testing. *Applied Psychological Measurement*, 22, 259-270.
- Vispoel, W. P., Rocklin, T. R., & Wang, T. (1994). Individual differences and test administration procedures: A comparison of fixed-item, computerized-adaptive, and self-adapted testing. *Applied Measurement in Education*, 53, 53-79.
- Vispoel, W. P., Wang, T., & Bleiler, T. (1997). Computerized adaptive and fixed-item testing of music listening skill: A comparison of efficiency, precision, and concurrent validity. *Journal of Educational Measurement*, 34, 43-63.
- Wainer, H., & Kiely, G. L. (1987). Item clusters and computerized adaptive testing: A case for testlets. *Journal of Educational Measurement*, 24, 185-201.
- Weiss, D. J. (1974). *Strategies of adaptive ability measurement* (Research Report 74-5). Minneapolis: Psychometric Methods Program, Department of Psychology, University of Minnesota.