

Methods and Models for the Construction of Weakly Parallel Tests

Jos J. Adema
University of Twente

Several methods are proposed for the construction of weakly parallel tests [i.e., tests with the same test information function (TIF)]. A mathematical programming model that constructs tests containing a prespecified TIF and a heuristic that assigns items to tests with information functions that are equal are important components of these methods. Numerical examples demonstrate

that tests can be constructed quickly and that the heuristic produces good results. However, the heuristic is not applicable for every set of practical constraints (e.g., constraints with respect to test administration time, test composition, or dependencies between items). *Index terms: item banking, heuristics, mathematical programming, test construction, weakly parallel tests.*

An item bank is a collection of items calibrated under an item response model. Given an item bank, tests can be constructed automatically by the application of mathematical programming models (Adema & van der Linden, 1989; Baker, Cohen, & Barmish, 1988; Boekkooi-Timminga, 1989; Theunissen, 1985; van der Linden & Boekkooi-Timminga, 1989). The goal of mathematical programming models is to optimize an objective function under a number of constraints. For tests constructed using mathematical programming models, the goal is to maximize the quality of the test under the constraint that the composition of the test must meet the requirements of the test constructor. A measure of the quality of a test is the test information function (TIF), because the information function for a maximum likelihood estimator of ability is the reciprocal of the (asymptotic) sampling variance of the estimator (Lord, 1980). Therefore, the TIF plays an important role in most mathematical programming models for test construction. Under the condition of local independence, the information function of a test can be computed by addition of the item information functions:

$$I(\theta) = \sum_{i=1}^N I_i(\theta) \quad , \quad (1)$$

where N is the number of items in the test,

θ is the ability parameter, and

$I_i(\theta)$ is the information function of item i .

Boekkooi-Timminga (1987, 1990) proposed mathematical programming models for the construction of weakly parallel tests. According to Samejima (1977), two weakly parallel test forms are "a pair of tests which measure the same ability and whose test information functions are identical" (p. 194). Weakly parallel tests measure the same variable equally well at all θ levels.

This paper presents two methods that construct weakly parallel tests based on a prespecified information function. A method is then described for selecting weakly parallel tests that are optimal with respect to the Maximin criterion. Numerical examples are presented that address the practicality of the proposed methods.

The Construction of Weakly Parallel Tests With a Prespecified Information Function

A Minimax (MI) Model for the Construction of Weakly Parallel Tests

The TIF is considered at a number of θ levels— θ_k , $k = 1, \dots, K$. The test constructor can select the number and spacing of these θ levels. Let $I_i(\theta_k)$ be the information value of item i at θ_k . Let $T(\theta_k)$ be the test information required at θ_k . The decision variables x_i in the mathematical programming model, denoting if an item is or is not selected for the test, are defined as follows:

$$x_i = \begin{cases} 0, & \text{item } i \text{ not in the test} \\ 1, & \text{item } i \text{ in the test,} \end{cases}$$

$$i = 1, \dots, I. \quad (2)$$

The decision variables u_k denote the difference between $T(\theta_k)$ and the TIF value at θ_k when the latter is smaller than $T(\theta_k)$:

$$u_k = \max [0, T(\theta_k) - \sum_{i=1}^I I_i(\theta_k)x_i] \quad (3)$$

If the TIF value at θ_k is larger than $T(\theta_k)$, the difference between the two is given by v_k :

$$v_k = \max [0, \sum_{i=1}^I I_i(\theta_k)x_i - T(\theta_k)] \quad (4)$$

The decision variable w is equal to the absolute value of the largest difference in the information function value over all θ levels:

$$w = \max_k (u_k + v_k) = \max (\max_k u_k, \max_k v_k) \quad (5)$$

Given these definitions of the decision variables, the MI model is formulated as follows:

$$\text{minimize } w \quad (6)$$

subject to

$$u_k + v_k \leq w, \quad k = 1, \dots, K, \quad (7)$$

$$\sum_{i=1}^I I_i(\theta_k)x_i + u_k - v_k = T(\theta_k), \quad k = 1, \dots, K, \quad (8)$$

$$\sum_{i=1}^I a_{ij}x_i = b_j, \quad j = 1, \dots, J, \quad (9)$$

$$\sum_{i=1}^I x_i = N \quad (10)$$

$$x_i \in \{0,1\}, \quad i = 1, \dots, I, \quad (11)$$

$$u_k, v_k \geq 0, \quad k = 1, \dots, K, \quad (12)$$

and

$$w \geq 0 \quad (13)$$

The objective of the model is to minimize the largest difference for all θ_k s between the given TIF and

the information function of the test to be constructed. The objective function (Equation 6) and the constraints (Equation 7) imply that the decision variable w is equal to the largest difference between $T(\theta_k)$ and the TIF at θ_k for all θ_k . The constraints in Equation 8 show that $u_k - v_k$ is equal to $T(\theta_k)$ minus the TIF value at θ_k . Equation 9 specifies the general notation for practical constraints, such as constraints due to the composition of the test or the administration time of the test (see van der Linden & Boekkooi-Timminga, 1989). The number of items in the test is equal to N by the constraint of Equation 10.

The Minimax Method

P weakly parallel tests can be constructed by applying Equations 6–13 P times. Every time the model is applied, the selected items are deleted from the item bank so that no item is contained in more than one test. A disadvantage of this method, as evidenced in the numerical examples below, is that sometimes the psychometric quality of successive tests decreases.

The Minimax Method and the Heuristic of Minimization of Differences (MIDI)

This two-stage method employs MI and the heuristic DIFMIN (minimization of differences) to construct P weakly parallel tests with a prespecified TIF. First, one large test is constructed that is P times the size of each of the tests to be constructed. Second, the items selected in Step 1 are assigned to the P tests so that the selected tests have the same composition and TIF. In this second step, DIFMIN is applied. In this case, the heuristic is used when the item bank must be partitioned due to certain practical constraints. For example, a mathematics item bank can be partitioned into geometry and intermediate algebra items—it may be determined that a test should contain 20 geometry items and 20 algebra items. For some constraints, DIFMIN is not applicable.

DIFMIN is based on known heuristics for solving the “makespan” scheduling problem (Coffman, Lueker, & Rinnooy Kan, 1988). In this problem, jobs must be distributed among two or more machines so that the time needed to process them is minimized. According to probabilistic analysis (e.g., Bruno & Downey, 1986; Frenk & Rinnooy Kan, 1987; Loulou, 1984), a good heuristic for the makespan scheduling problem is “largest processing time” (LPT).

In LPT, jobs are sorted in order of decreasing processing time and each job is assigned to the machine for which the sum of the processing times of the jobs already assigned is smallest. In the context of test construction, tests can be seen as “machines” and their items as “jobs.” Each item, however, has K characteristics (the information function values at θ_k , $k = 1, \dots, K$). Because this problem involves K characteristics, practical constraints, and the requirement of equal numbers of items, the construction of weakly parallel tests is more difficult than the “makespan” scheduling problem.

As in LPT, the items are sorted for each θ_k level in order of decreasing information function value. Then, the items are assigned to the tests on a one-by-one basis. In each iteration of DIFMIN, items are assigned to a specific test under the restriction that the items must be from the same subset in the item bank. The goal for each item assignment is to reduce the largest difference in the information functions for all tests and θ levels.

In the following description of the heuristic, Q denotes the Q th iteration. After N iterations, DIFMIN stops.

Step 1. Use MI (Equations 6–13) to construct a test that is P times the size of each of the P tests to be constructed with $P \cdot T(\theta_k)$ instead of $T(\theta_k)$ in Equation 8, $P \cdot N$ instead of N in Equation 10, and $P \cdot b_j$ instead of b_j in Equation 9.

Step 2. Apply the heuristic DIFMIN.

Step 2a. For each θ_k , sort the items selected in Step 1 in order of decreasing information value.

Step 2b. Initialization: The first item in the list of θ_1 is assigned to the first test ($p = 1$). Set Q equal to 1.

Step 2c. Complete the following for tests $p = 2$ to P successively: Select the item having the most information at θ_1 that belongs to the same part of the item bank as the item selected in Step 2b, but not already assigned, and assign it to test p .

Step 2d. If the number of items in each test is equal to N (i.e., if $Q = N$), then stop; otherwise, let $Q = Q + 1$ and compute

$$\max(k) = \max_p S_p(\theta_k), \quad k = 1, \dots, K, \quad (14)$$

where $S_p(\theta_k)$ is the sum of the information function values at θ_k of the items already selected for test p . Compute θ_{min} and test p_{min} , where the difference in information function values with respect to all tests and θ levels is largest:

$$\max(\min) - S_{p_{min}}(\theta_{min}) = \max_p \max_k [\max(k) - S_p(\theta_k)] \quad . \quad (15)$$

The first item in the list of θ_{min} not yet assigned to a test is assigned to p_{min} .

Step 2e. Compute θ_{min} and p_{min} such that

$$\max(\min) - S_{p_{min}}(\theta_{min}) = \max_{p \in \pi_{Q-1}} \max_k [\max(k) - S_p(\theta_k)] \quad , \quad (16)$$

where π_{Q-1} is the set of tests with $Q - 1$ items. Select the first item in the list of θ_{min} that has not already been selected for a test and belongs to the same part of the item bank as the last item selected in Step 2d. Repeat this step until all tests contain Q items.

Step 2f. Go to Step 2d.

Example of the Heuristic

Suppose 6 items (see Table 1) are selected in Step 1 and that these items are to be assigned to two tests such that they have (approximately) the same TIF at θ_1 and θ_2 . Step 2a for these items results in the following order of items at the two θ levels: for θ_1 , items 3,6,2,1,4,5; and for θ_2 , items 5,6,2,4,1,3.

Table 1
Information Values at Two θ Levels and Subset Membership
(G = Geometry, IA = Intermediate Algebra) for Six Test Items

θ level and Subset	Item					
	1	2	3	4	5	6
θ_1	.8	1.1	2.0	.7	.5	1.4
θ_2	.6	.9	.4	.8	1.2	1.1
Subset	G	G	G	G	IA	IA

According to Step 2b, Item 3 is assigned to Test 1, and Item 2 is assigned to Test 2. Item 6 (Step 2c), however, is not assigned to Test 1, because Items 3 and 6 belong to different subsets. The computations in Step 2d imply that the most informative item at θ_1 should be selected for Test 2. Item 3 was already selected; thus, Item 6 was selected for Test 2. Because Item 5 is the only item belonging to the same subset as Item 6, this item was selected for Test 1 (Step 2e). After execution of Step 2e, Step 2d is executed again. The next item to be assigned should give the most information at θ_2 and is selected for Test 1. This implies that Item 4 is assigned to Test 1. Only Item 1 remains, and it is

assigned to Test 2. The results are as follows:

Test	Items	$I(\theta_1)$	$I(\theta_2)$
1	3,4,5	3.2	2.4
2	1,2,6	3.3	2.6

DIFMIN, as presented here, can be changed and extended in several ways. For example, the choice of θ_1 in Step 2c is arbitrary; any value of θ can be used. The number of items to be selected from a part of the item bank could be given as a lower or upper bound rather than as an exact number. In that case, DIFMIN cannot always find an item from the same part of the item bank (Step 2e).

Ackerman (1989) introduced another modification to the heuristic. He did not take the practical constraints into account in selecting items, but checked the content of each item after it was selected. If the specific content area of the item was already represented in the test, the next best item is selected. This process is repeated until an item is found for which the content area had not reached the prespecified number of items.

The Construction of Weakly Parallel Tests Under the Maximin Criterion

This method is based on MIDI and constructs tests that are optimal with respect to the Maximin (MA) criterion. This criterion states that a test should give as much information as possible, but it operates under the restrictions that the TIF has the shape required by the test constructor and that the number of items is fixed. This criterion was proposed by van der Linden and Boekkooi-Timminga (1989). In contrast to the previous method, the test constructor specifies a relative instead of an absolute target TIF.

Maximin Model

The TIF is considered at a number of ability levels, θ_k , $k = 1, \dots, K$. The number and spacing of these levels should be specified by the test constructor. The relative shape of the target TIF should be specified by the test constructor by selecting r_k , $k = 1, \dots, K$. Let y be a decision variable such that (r_1y, \dots, r_Ky) is a series of lower bounds to the TIF. If N is the prescribed number of items in the test, then the MA model is formulated as follows:

$$\text{maximize } y, \tag{17}$$

subject to

$$\sum_{i=1}^I I_i(\theta_k) x_i - r_k y \geq 0, \quad k = 1, \dots, K, \tag{18}$$

$$\sum_{i=1}^I x_i = N, \tag{19}$$

$$x_i \in \{0,1\}, \quad i = 1, \dots, I, \tag{20}$$

and

$$y \geq 0. \tag{21}$$

The lower bounds are forced to be as high as possible by maximizing y in the objective function (Equation 17). Equation 18 implements the constraints on the decision variable. The number of items in the test is controlled by Equation 19.

MADI Method

The MADI method combines the MA model and the heuristic DIFMIN for the construction of weakly parallel tests that are optimal with respect to the MA criterion. The methods MIDI and MADI differ only in the criterion used (MI for the method MIDI and MA for the method MADI).

Step 1. Construct a test using the MA model that is P times the size of each of the weakly parallel tests.

Step 2. Select items for the large test and assign them to the P tests by the heuristic DIFMIN, such that weakly parallel tests are created.

Numerical Examples

In the examples below, mathematical programming models were solved several times. A heuristic was used because the application of exact algorithms is time consuming. The applied heuristic solution strategy for the MI and MA models is called optimal rounding (van der Linden & Boekkooi-Timminga, 1989), because the practical constraints in the examples are such that a feasible sub-optimal solution always will be found. Optimal rounding in this sense means that first the relaxed model [$0 \leq x_i \leq 1$ instead of $x_i \in (0,1)$] is solved, then all variables x_i equal to 0 or 1 are fixed; finally, the reduced model is solved to optimality by the branch-and-bound method (Land & Doig, 1960). In the examples, optimal rounding—a rounding procedure that provides feasible solutions for the practical constraints—was used. For other kinds of practical constraints, optimal rounding does not always give a feasible solution. In that case, the heuristic described by Adema, Boekkooi-Timminga, and van der Linden (1991) could be used. The execution times of the programs are shown in the tables to illustrate the practicality of the methods when executed on an IBM 9370 computer.

The item bank used in the examples was an American College Testing (ACT) item bank described by Ackerman (1989). The bank consisted of 600 items; 520 items were from 13 previously administered ACT Assessment Program tests and 80 were from the Collegiate Mathematics Placement Program. The items were calibrated using the three-parameter IRT model (Birnbaum, 1968). The bank was divided into six content areas: Arithmetic and Algebraic Operations (AAO), Arithmetic and Algebraic Reasoning (AAR), Geometry (G), Intermediate Algebra (IA), Number and Numeration Concepts (NNS), and Advanced Topics (AT). Items were selected to create weakly parallel tests with 40 items (4 AAO items, 14 AAR items, 8 G items, 8 IA items, 4 NNS items, and 2 AT items).

Methods MI and MIDI

The θ levels and the target information function values [$\theta_k, T(\theta_k)$] used were (-1.6, 2.0), (-.8, 5.4), (0.0, 12.1), (.8, 21.3), (1.6, 10.8), and (2.4, 3.1), respectively. Six tests were constructed. Table 2 shows the differences from the target values for the MI method, and Figure 1 shows the six TIFs. After the first five tests were constructed, there was a lack of good items. Therefore, the last TIF was not as close to the target as those of the other tests.

In Table 3, the TIFs of the tests constructed by MIDI are shown for P ranging from 2 to 6. Figure 2 shows the TIFs for $P = 6$. The item bank contained a few items with extremely high difficulty. Therefore, it was possible that the TIFs would be high for $\theta > 0$. The tests were constructed simultaneously by MIDI and, therefore, the tests show no decrement in quality in the order of selection. Most of the computing time was needed for the first step in the method. This step was more time consuming for $P = 6$ than for P ranging from 2 to 5. Thus, the total computing time was larger for $P = 6$.

Table 2
Differences From the Target for Weakly Parallel
Tests Constructed by Method MI
(Computing Time Was 1.97 Minutes)

<i>P</i>	θ					
	-1.6	-0.8	0.0	0.8	1.6	2.4
1	.1	.1	0.0	.2	.2	.1
2	.2	.1	.1	.1	0.0	.2
3	0.0	.4	.4	.3	.4	0.0
4	.1	0.0	0.0	.1	.1	.1
5	.1	0.0	.2	.1	.1	.2
6	.5	.6	.9	.9	.5	.1

Method MADI

The θ levels and the relative information function values (θ_k, r_k) used were (-1.6, 2.0), (-0.8, 5.4), (0.0, 12.1), (0.8, 21.3), (1.6, 10.8), and (2.4, 3.1), respectively. The TIF values are shown in Table 4 for $P = 2$ through $P = 6$, and the TIFs for $P = 6$ are displayed in Figure 3. Results show that the TIFs for MADI (maximizing relative information) were higher than for MI and MIDI (absolute information).

Discussion

MI and MIDI can be used for the construction of weakly parallel tests with a fixed target TIF. Both methods can construct tests in a few minutes from a bank of 600 items. Thus, computing time is not a problem when using the methods in a noninteractive mode. A drawback of MI is that tests are selected sequentially, so that tests constructed later tend not to be as good as earlier constructed tests.

Figure 1
Information Functions of Tests Constructed by Method MI for $P = 6$

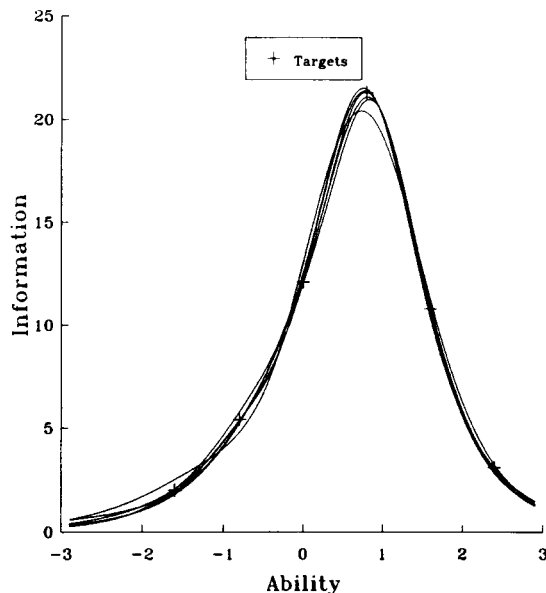


Table 3
 Differences From the Target for Weakly Parallel Tests
 Constructed by MIDI, and Computing Time in Minutes

	θ					Time
	-1.6	-.8	0.0	.8	1.6	
<i>P</i> = 2						
0.0	.3	.1	.2	0.0	0.0	.55
0.0	0.0	.1	.2	0.0	0.0	
<i>P</i> = 3						
.1	.2	.2	.2	.1	0.0	.63
0.0	.2	.1	0.0	0.0	.2	
.1	0.0	0.0	.2	.1	.1	
<i>P</i> = 4						
0.0	.1	.1	0.0	.1	.1	.61
.1	.1	.1	.1	.2	0.0	
.2	0.0	.2	.1	.1	.1	
.1	.1	.1	0.0	.1	0.0	
<i>P</i> = 5						
.1	0.0	.1	.1	.1	0.0	.63
.1	.1	0.0	.1	0.0	.1	
0.0	0.0	0.0	0.0	.1	.1	
0.0	.1	.1	0.0	.1	0.0	
0.0	.1	.1	.1	0.0	.1	
<i>P</i> = 6						
.1	.1	0.0	.1	0.0	0.0	.86
.1	.1	0.0	.1	.1	0.0	
.2	0.0	.1	.3	.1	.4	
.1	0.0	0.0	.1	.3	0.0	
.1	0.0	0.0	.1	0.0	0.0	
.1	.2	.4	0.0	.2	.1	

Figure 2
 Information Functions of Tests Constructed by Method MIDI for *P* = 6

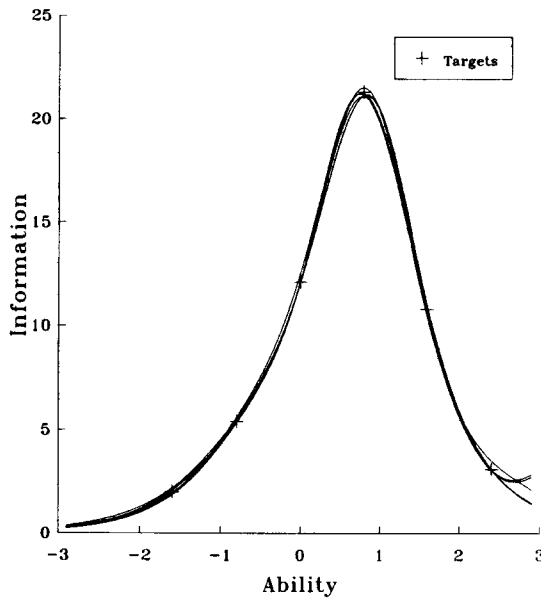
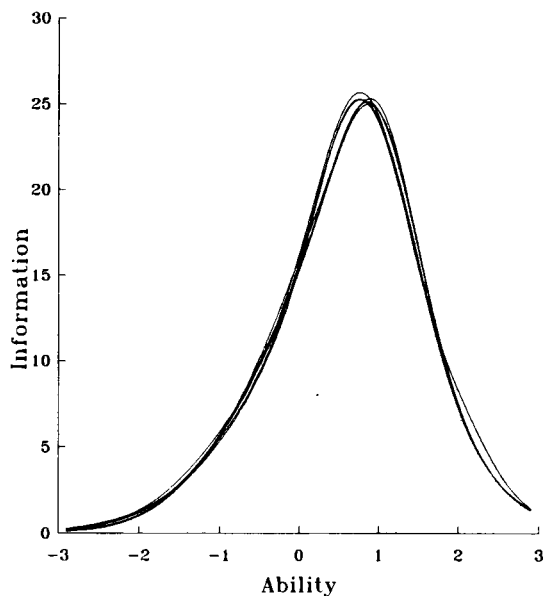


Table 4
Information Functions for Weakly Parallel Tests
Constructed by MADI, and Computing Time in Minutes

	θ					Time
	-1.6	-.8	0.0	.8	1.6	
<i>P</i> = 2						
3.1	8.7	18.2	31.9	16.6	5.0	.63
2.9	8.8	18.4	32.0	15.6	5.1	
<i>P</i> = 3						
2.8	7.8	17.3	30.4	15.4	4.6	.61
2.7	7.9	17.2	30.0	15.8	4.4	
3.0	7.9	16.9	30.0	15.4	4.5	
<i>P</i> = 4						
2.9	7.5	16.3	28.2	14.6	4.2	.65
2.5	7.4	16.7	28.9	14.6	4.4	
2.5	7.2	15.7	28.0	15.6	3.8	
2.7	7.1	15.9	28.0	14.6	4.7	
<i>P</i> = 5						
2.5	6.8	15.9	27.5	14.1	4.7	.65
2.5	7.2	15.6	26.5	14.4	3.6	
2.8	7.5	15.6	26.5	14.5	4.2	
2.3	7.2	15.6	26.5	14.3	3.6	
2.5	6.9	15.3	26.5	15.2	3.6	
<i>P</i> = 6						
2.3	6.9	15.8	25.3	13.9	3.5	.70
2.4	7.2	15.4	25.2	14.5	3.5	
2.7	7.2	16.2	25.8	13.7	4.4	
2.4	7.1	15.3	25.0	14.5	3.5	
2.2	7.3	16.0	25.2	14.4	3.5	
2.4	6.7	15.4	25.3	13.7	3.5	

Figure 3
Information Functions of Tests Constructed by Method MADI for *P* = 6



This problem occurred for $P = 6$ (see Table 2). The information function of the sixth test was not as good a match to the target TIF as were the previous five tests.

For MIDI, the TIFs were closer to the target TIF than for the MI method, because the tests were constructed simultaneously in the MIDI method. MIDI is also faster than MI. MIDI uses the heuristic DIFMIN only for the case in which practical constraints imply a partition of the item bank. A problem with MIDI is that it is not applicable for other types of practical constraints. MI does not have this problem and gives good results as long as the item bank is not exhausted ($P \leq 5$ in the example).

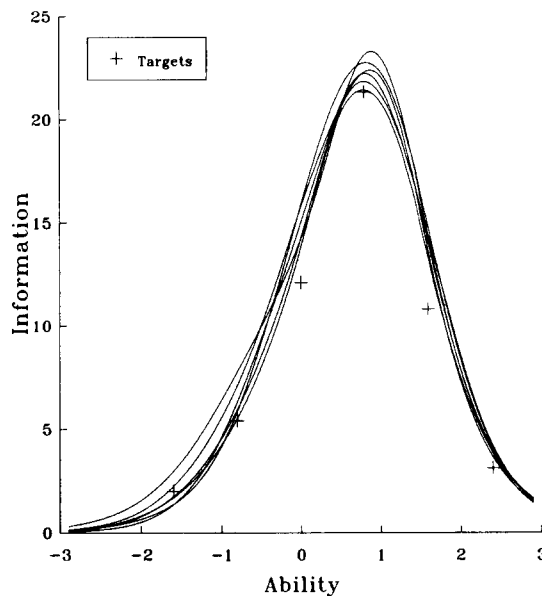
Ackerman (1989) proposed a heuristic for creating parallel tests, and used the same item bank in his numerical examples. Figure 4 shows his results. Comparison of Figures 1, 2, and 4 indicate that methods MI and MIDI produce tests of higher quality than Ackerman's heuristic.

In all cases, MADI constructed weakly parallel tests in less than a minute. The method suffers the same drawback as MIDI: It is not applicable to all kinds of constraints.

Figure 4

Information Functions of Tests Constructed by Ackerman (1989)

(From *An alternative methodology for creating parallel test forms using the IRT information function* by T. A. Ackerman, 1989. Paper presented at NCME annual meeting, San Francisco CA, U.S.A. Reprinted by permission.)



References

- Ackerman, T. A. (1989, March). *An alternative methodology for creating parallel test forms using the IRT information function*. Paper presented at NCME annual meeting, San Francisco.
- Adema, J. J., & van der Linden, W. J. (1989). Algorithms for computerized test construction using classical parameters. *Journal of Educational Statistics, 14*, 279–290.
- Adema, J. J., Boekkooi-Timminga, E., & van der Linden, W. J. (1991). Achievement test construction using 0-1 linear programming. *European Journal of Operational Research, 55*, 103–111.
- Baker, F. B., Cohen, A. S., & Barmish, B. R. (1988). Item characteristics of tests constructed by linear programming. *Applied Psychological Measurement, 12*, 189–199.
- Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick, *Statistical theories of mental test scores* (pp. 397–479). Reading MA: Addison-Wesley.
- Boekkooi-Timminga, E. (1987). Simultaneous test construction by zero-one programming. *Methodika, 1*, 101–112.
- Boekkooi-Timminga, E. (1989). *Models for computerized test construction*. Doctoral dissertation. De Lier, The Netherlands: Academisch Boeken Centrum.
- Boekkooi-Timminga, E. (1990). The construction of parallel tests from IRT-based item banks. *Journal of Educational Statistics, 15*, 129–145.
- Bruno, J. L., & Downey, P. J. (1986). Probabilistic bounds on the performance of list scheduling. *SIAM Journal on Computing, 15*, 409–417.
- Coffman, E. G., Jr., Lueker, G. S., & Rinnooy Kan, A. H. G. (1988). Asymptotic methods in the probabilistic analysis of sequencing and packing heuristics. *Management Science, 34*, 266–290.
- Frenk, J. B. G., & Rinnooy Kan, A. H. G. (1987). The asymptotic optimality of the LPT rule. *Mathematics of Operations Research, 12*, 241–254.
- Land, A. H., & Doig, A. G. (1960). An automated method for solving discrete programming problems. *Econometrica, 20*, 497–520.
- Lord, F. M. (1980). *Applications of item response theory to practical testing problems*. Hillsdale NJ: Erlbaum.
- Loulou, R. (1984). Tight bounds and probabilistic analysis of two heuristics for parallel processor scheduling. *Mathematics of Operations Research, 9*, 142–150.
- Samejima, F. (1977). Weakly parallel tests in latent trait theory with some criticisms of classical test theory. *Psychometrika, 42*, 193–198.
- Theunissen, T. J. J. M. (1985). Binary programming and test design. *Psychometrika, 50*, 411–420.
- Van der Linden, W. J., & Boekkooi-Timminga, E. (1989). A maximin model for test design with practical constraints. *Psychometrika, 54*, 237–247.

Acknowledgments

The author thanks Sjoerd Baas and Noud Gademann for their helpful comments, ACT and Terry Ackerman for making the item bank available, and Lorette Bosch for her assistance in preparing the figures.

Author's Address

Send requests for reprints or further information to Jos J. Adema, University of Twente, Department of Education, P.O. Box 217, 7500 AE Enschede, The Netherlands.