DOCUMENT RESUME

ED 326 581 TM 015 953

AUTHOR	Adema, Jos J.
TITLE	Methods and Models for the Construction of Weakly Parallel Tests. Research Report 90-4.
INSTITUTION	Twente Univ., Enschede (Netherlands). Dept. of Education.
PUB DATE	Sep 90
NOTE	44p.; For a related document, see TM 015 952.
AVAILABLE FROM	Bibliotheek, Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.
PUB TYPE	Reports - Evaluative/Feasibility (142)
EDRS PRICE	MF01/PC02 Plus Postage.
DESCRIPTORS	*College Entrance Examinations; *Computer Assisted Testing; Equations (Mathematics); Foreign Countries; Heuristics; Higher Education; *Item Banks; Item Response Theory; *Mathematical Models; Mathematics Tests; Student Placement; *Test Construction
IDENTIFIERS	*Information Function (Tests); Mathematical Programing; Maximin Model; *Parallel Test Forms

ABSTRACT

Methods are proposed for the construction of weakly parallel tests, that is, tests with the same test information function. A mathematical programing model for constructing tests with a prespecified test information function and a heuristic for assigning items to tests such that their information functions are equal play an important role in the methods. The MI and MIDI methods are proposed for constructing tests with a prespecified test information function applying the Minimax model. Similar methods, MAMI and MADI, are provided for construction of a weakly parallel test approximately equal with respect to the Maximin criterion. The four methods were applied on a real item bank of 600 items from college placement mathematics tests (520 items were from 13 previously administered American College Testing Assessment Program tests, and 80 were from the Collegiate Mathematics Placement Program). The numerical examples indicated that the tests were constructed quickly and that the heuristic gave good results. However, the heuristic was not applicable for every set of practical constraints (i.e., constraints with respect to test administration time, test composition, or dependencies between items). Four tables and four graphs present information about the constructed tests. (Author/SLD)

Methods and Models for the Construction of Weakly Parallel Tests

Research Report 90-4

Jos J. Adema

ment of EDUCATION

Division of Educational Measurement and Data Analysis

University of Twente

Colofon:
Typing: L.A.M. Bosch-Padberg
Cover design: Audiovisuele Sectie TOLAB
Toegepaste Onderwijskunde
Printed by: Centrale Reproductie-afdeling

Methods and Models for the Construction of Weakly Parallel Tests

Jos J. Adema

Methods and models for the construction of weakly parallel tests , Jos J. Adema - Enschede : University of Twente, Department of Education, September, 1990. - 36 pages

Abstract

Computerized test construction by mathematical programming is possible given the availability an item bank calibrated under an IRT model. In this paper methods are proposed for the construction of weakly parallel tests, that is, tests with the same test information function. In the methods a mathematical programming model for constructing tests with a prespecified test information function and a heuristic for assigning items to tests such that their information functions are equal play an important role. Numerical examples show that the tests are constructed very fast and that the heuristic gives good results. However, the heuristic is not applicable for every set of practical constraints, such as, constraints with respect to test administration time, test composition, or dependencies between items.

Keywords: Item Banking, Test Construction, Weakly Parallel Tests, Mathematical Programming, Heuristics.

Methods and Models for the Construction of Weakly Parallel Tests

In this paper an item bank is defined as a large 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; van der Linden & Boekkooi-Timminga, 1989; Theunissen, 1985). The goal of mathematical programming models is to optimize an objective function under a number of constraints. In mathematical programming models for test construction the goal is often to maximize the quality of the test under the constraint that the composition of the test is as required by the test constructor. A measure for the quality of a test is the test information function, 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 test information function 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:

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

where N is the number of items in the test, θ the ability parameter, and $I_i(\theta)$ the information function of item i.

Boekkooi-Timminga (1987, 1990) has proposed mathematical programming models for the construction of weakly parallel tests, where 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".

Two main parts of this paper can be distinguished. In Part A it is assumed that a test information function is available and one or more tests should be constructed such that all tests have the same information function. In the following section two methods are described for solving this problem. In Part B of the paper two methods are given for selecting weakly parallel tests that are optimal with respect to the Maximin criterion. In this part first the criterion is formulated. Then, two methods are proposed for solving the problem. Next, some numerical examples are worked out to give an impression of the practicality of the proposed methods with respect to CPU time and accuracy.

PART A: The Construction of Weakly Parallel Tests with a Prespecified Information Function

In this part it is shown how one or more tests with a prespecified test information function can be constructed. In the first section the case of one test is considered. In the last two sections methods are given for the construction of more than one test. These methods make use of the Minimax Model that is presented in the first section.

A Minimax Model for the Construction of Weakly Parallel Tests

In this section a mathematical programming model for the selection of items is given. It produces tests that approximate a prespecified test information function best and in addition guarantees that, for example, the number of items in the test and the composition of the test are as specified by the test constructor. Such a model can be used to select a test that is weakly parallel to a given test.

The test information function is considered at a number of ability levels θ_k, $k = 1,.., K$. The test constructor can choose the number and spacing of these ability levels. Let $I_i(\theta_k)$ be the information value of item i at ability level θ_k and $T(\theta_k)$ the test information required at ability level 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:

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

The decision variables u_k denote the difference between $T(\theta_k)$ and the test information function value at θ_k for the case that the latter is smaller than $T(\theta_k)$:

$$u_k = \max_k \{0,\ T(\theta_k) - \sum_{i=1}^{I} I_i(\theta_k)x_i\}.$$

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

$$v_k = \max_k \{0,\ \sum_{i=1}^{I} I_i(\theta_k)x_i - T(\theta_k)\}.$$

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

$$w = \max_k \{\ u_k + v_k\ \}$$

$$= \max \{\max_k u_k;\ \max_k v_k\}.$$

Given these definitions of the decision variables, the Minimax Model is formulated as:

(1) minimize w,

subject to

(2) $u_k + v_k \leq w,$ $k = 1,.., K,$

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

(4) $\sum_{i=1}^{I} a_{ij}x_i = b_j,$ $j = 1,.., J,$

(5) $\sum_{i=1}^{I} x_i = N,$

(6) $x_i \in \{0,1\},$ $i = 1,.., I,$

(7) $u_k, v_k \geq 0,$ $k = 1,.., K,$

(8) $w \geq 0.$

The objective of the model is to minimize the largest difference for all θ_k between the given test information function and the information function of the test to be constructed. The objective function (1) and the constraints (2) imply that the decision variable w is equal to the largest difference between $T(\theta_k)$ and the test information function at ability level θ_k for all θ_k. By constraints (3) $u_k - v_k$ is equal to $T(\theta_k)$ minus the test information function value at θ_k. The constraints presented in (4) are a general notation for practical constraints, that is, constraints with respect to the composition of the test, the administration time of the test, etc. (see van der Linden & Boekkooi-Timminga, 1989). The number of items in the test is set equal to N by constraint (5).

Method MI

The construction of, say, P weakly parallel tests can be done by applying model (1)-(6) P times. Every time the model has been applied, the selected items are deleted from the item bank so that no item is contained in more than one test. Throughout this paper, this method of constructing weakly parallel tests will be called MI. As can be seen in the numerical examples below sometimes the psychometric quality of the tests will decrease in the order in which they are constructed by method MI. In the next section a method is described that does not have this drawback.

Method MIDI

This section is addressed to the problem of simultaneous construction of P weakly parallel tests with a prespecified test information function. The problem is solved here in two steps that will now be described. In the first step one large test is constructed that is P times the size of each of the tests to be constructed. In the second step the items are assigned to the P tests such that the selected tests have the same composition and test information function. In the second step the heuristic DIFMIN is applied. This heuristic will be described here for the case of practical constraints that imply a partition of the item bank. Consider for example, a mathematical item bank that can be partitioned into geometry and intermediate algebra items; one may determine that a test should contain 20 geometry and 20 intermediate algebra items. For other types of constraints DIFMIN should be adapted or is not applicable.

DIFMIN is based on known heuristics for solving the "makespan" scheduling problem (e.g., Coffman, Lueker & Rinnooy Kan, 1988). In this problem jobs have to be distributed among two or more parallel machines so as to minimize the time needed to process them. A good heuristic for the "makespan" scheduling problem according to probabilistic analysis (see, e.g., Bruno & Downey, 1986; Frenk & Rinnooy Kan, 1985; Loulou, 1984) is LPT (Largest Processing Time). In LPT the jobs are sorted in order of decreasing processing time and, next, in decreasing order of processing time, each job is assigned to the machine for which the sum of the processing times of the jobs already assigned is smallest. In our problem the tests can be seen as machines and the items as jobs. Each item, however, does not have one characteristic but K, namely the information function values at θ_k, k = 1,.., K. This along with the practical constraints and the fact that the number of items should be equal for each test make the construction of weakly parallel tests more difficult than the "makespan" scheduling problem. As in LPT the heuristic starts with sorting the items, that is, for each ability level θ_k the items are sorted 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 to each test an item is assigned under the restriction that these items be from the same subset of the partition of the item bank. The goal for each assignment is to close the largest gap in information function value for all tests and ability levels.

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

Step 1: Use Minimax (1) - (8) to construct a test that is P times the size of each of the P tests to be constructed with $P*T(\theta_k)$ instead of $T(\theta_k)$ in (3), P*N instead of N in (5) and $P*b_j$ instead of b_j in (4).

Step 2 (Heuristic DIFMIN)

Step 2a: For each ability level, 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: Successively for tests p = 2 to P: Choose the item having the most information at θ_1, belonging to the same part of the item bank as the item selected in Step 2b, and not already assigned, and assign it to test p.

Step 2d: If the number of items in each test is equal to N, that is, if Q = N, then STOP, else let Q := Q+1 and compute

$$\max(k) = \max_{p} I_p(\theta_k), \qquad k = 1,.., K,$$

where $I_p(\theta_k)$ is the sum of the information function values at θ_k of the items already selected for test p.

Compute the ability level θ_{min} and test p_{min} which has the largest gap in information function value with respect to all tests and ability levels:

$$\max(\min) - I_{p_{min}}(\theta_{min}) = \max_{p} \max_{k} \{\max(k) - I_p(\theta_k)\}.$$

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

$$\max(\min) - I_{p_{min}}(\theta_{min}) = \max_{p \in \pi_{Q-1}} \max_{k} \{\max(k) - I_p(\theta_k)\}$$

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:

Suppose 6 items are selected in Step 1 and that these items should be assigned to two tests such that they have (approximately) the same test information function at ability levels θ_1 and θ_2. The data are given by:

Item	1	2	3	4	5	6
$I(\theta_1)$	0.8	1.1	2.0	0.7	0.5	1.4
$I(\theta_2)$	0.6	0.9	0.4	0.8	1.2	1.1
Subset	G	G	G	G	IA	IA

where G stands for geometry and IA for intermediate algebra. The following lists are the result of Step 2a

θ_1	3	6	2	1	4	5
θ_2	5	6	2	4	1	3.

According to Step 2b item 3 is assigned to test 1. Also, item 2 is assigned to test 2 and not item 6 (Step 2c), 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 is selected for test 2. Because item 5 is the only item belonging to the same subset as item 6, this item is 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 ability level θ_2 and is selected for test 1. This implies that item 4 is assigned to test 1. Only item 1 remain and it is assigned to test 2. The results are as follows:

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

PART B: The Construction of Weakly Parallel Tests under the Maximin Criterion

In this part two methods based on MI and MIDI are described for the construction of weakly parallel tests, where the tests are optimal with respect to the Maximin criterion. This criterion states that a test should give as much information as possible under the restriction that the test information function 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). As opposed to Part A, the test constructor now specifies a relative instead of an absolute target test information function. A brief introduction of the Maximin Model for the construction of one test will be given below.

Maximin Model

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

(9) maximize y,

subject to

(10) $$\sum_{i=1}^{I} I_i(\theta_k)x_i - r_ky \geq 0, \qquad k = 1,.., K,$$

(11) $$\sum_{i=1}^{I} x_i = N,$$

(12) $$x_i \in \{0,1\}, \qquad i = 1,.., I,$$

(13) $$y \geq 0.$$

The lower bounds are forced to be as high as possible by maximizing y in the objective function (9). The constraints (10) are formulated such that $(r_1y,..., r_Ky)$ is a series of lower bounds to the test information function. The number of items in the test is controlled by constraint (11).

In the next two sections methods are proposed for the construction of weakly parallel tests, which are optimal with respect to the Maximin criterion.

Method MAMI

If the test information function of the weakly parallel tests is known the tests can be selected by applying method MI. Method MAMI is based on the same idea. Two steps can be distinguished in the method. In the first step an estimate of the test information function is computed. In the second step method MI is applied.

Step 1: A good approximation of the test information function is computed by constructing a large test with the Maximin Model that is P times the size of each of the P tests. By dividing the information function values of the large test at the specified ability levels by P, a target test information function is found for the Minimax Model.

Step 2: By applying the Minimax Model P times each time deleting items selected at an earlier stage the P weakly parallel tests are constructed (Method MI).

Example:

An item bank consists of 400 items. The first 200 items are of the multiple-choice type and the other 200 items are of the matching type. A test constructor wants to select items for 3 weakly parallel tests each with 20 multiple-choice and 20 matching items. S(he) needs tests that measure well in the ability range [-2,2] and, therefore, the test information function is specified at the ability levels $\theta_1 = -2$, $\theta_2 = 0$, and $\theta_3 = 2$, and the corresponding values of r_k are $r_1 = r_2 = r_3 = 1$. The Maximin Model is formulated as:

maximize y,

subject to

$$\sum_{i=1}^{400} I_i(\theta_k)x_i - y \geq 0, \qquad k = 1, 2, 3,$$

$$\sum_{i=1}^{200} x_i = 60,$$

$$\sum_{i=201}^{400} x_i = 60,$$

$$x_i \in \{0,1\}, \qquad i = 1,.., 400,$$

$$y \geq 0.$$

Let the information function values of the large test be $L(\theta_k)$, $k = 1, 2, 3$; then the target test information function is $T(\theta_k) = L(\theta_k)/3$. Thus, the following Minimax Model should be solved three times:

$$\text{minimize } w,$$

subject to

$$u_k + v_k \le w, \qquad k = 1, 2, 3,$$

$$\sum_{i \in I} I_i(\theta_k)x_i + u_k - v_k = T(\theta_k), \qquad k = 1, 2, 3,$$

$$\sum_{i \in I_{MC}} x_i = 20,$$

$$\sum_{i \in I_M} x_i = 20,$$

$$x_i \in \{0,1\}, \qquad i \in I,$$

$$u_k, v_k \ge 0, \qquad k = 1, 2, 3,$$

$$w \ge 0.$$

Every time a test is constructed the items already selected are deleted from the item bank. Subset I is the set of items not deleted from the item bank. Subsets I_M and I_{MC} are sets of items of the matching and multiple-choice type not deleted from the item bank.

Method MADI

The difference between method MIDI and MADI is that in the latter not the Minimax Model but the Maximin Model is used in Step 1.

Step 1: As in method MAMI, method MADI starts with the construction of a test by the Maximin Model that is P times the size of each of the weakly parallel tests.

Step 2: Next, the items selected for the large test are assigned to the P tests by the heuristic DIFMIN such that weakly parallel tests are created.

Example:

In this example the same problem as in the last example is considered. First, a large test is constructed as in method MAMI. Heuristic DIFMIN then assigns the selected items to the tests such that all three tests consist of 20 multiple-choice and 20 matching items and the test information functions are approximately equal.

Numerical Examples

In the examples below mathematical programming models had to be solved several times. The application of exact algorithms is too time consuming, therefore a heuristic has been used. The applied heuristic solution strategy is in all cases so-called optimal rounding (van der Linden & Boekkooi-Timminga, 1989), because the practical constraints in the examples are such that always a feasible suboptimal solution will be found. Optimal rounding in this sense means, that first the relaxed model ($0 \le x_i \le 1$ instead of $x_i \in \{0,1\}$) is solved, then all variables x_i equal to 0 or 1 are fixed and, finally, the reduced model is solved to optimality by the branch-and-bound method (Land & Doig, 1960). The optimal rounding method was implemented in ECL control programs for the software package MPSX/370 V2. ECL is a computer language based on PL\1. All methods considered in this paper were implemented in control programs such that they could be executed in one run. The execution times of the control programs are displayed in the tables below to show the practicability of the methods. The runs were executed on an IBM9370 computer.

The item bank used in the examples is an existing Academic College Testing (ACT) item bank and was described by Ackerman (1989). The bank consisted of 600 items; 520 items were from 13 previously administered ACT Assessment Program (AAP) tests and 80 were from the Collegiate Mathematics Placement Program (CMPP). The items were calibrated using the 3-parameter model (Birnbaum, 1968). The bank was divided into six content areas:

(1) Arithmetic and Algebraic Operations (AAO);

(2) Arithmetic and Algebraic Reasoning (AAR);

(3) Geometry (G);

(4) Intermediate Algebra (IA);

(5) Number and Numeration Concepts (NNS);

(6) Advanced Topics.

From the bank 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 Minimax Model (1) - (8) can be applied to construct one or more weakly parallel tests with a fixed target test information function. The methods MI and MIDI described above have been developed for this case. In this section the ability levels and the target information function values (θ_k, $T(\theta_k)$) are (-1.6, 2.0), (-.8, 5.4), (.0, 12.1), (.8, 21.3), (1.6, 10.8), and (2.4, 3.1), respectively.

The results for MI are displayed in Table 1. The number of tests P is equal to 6. The test information function values are given for all the tests.

Insert Table 1 here

Figure 1 shows the 6 test information functions.

Insert Figure 1 here

After the first 5 tests were constructed there was a lack of "good" items. Therefore, the last test information function having the lowest top, is not as close to the target as the info ation functions of the other tests.

In Table 2 the test information functions of the tests constructed by MIDI are shown for P ranging from 2 to 6.

Insert Table 2 here

An illustration of the test information functions for P = 6 is given in Figure 2.

Insert Figure 2 here

The item bank contains a few items with extremely high difficulty. Therefore, it is possible again that the test information functions increase for ability levels outside the range considered. The tests were constructed simultaneously by MIDI and, therefore, there is no decrement in quality of the tests in the order of selection. Most of the CPU 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 CPU time was larger for P = 6.

Methods MAMI and MADI

Two methods have been described for the construction of weakly parallel tests which are optimal with respect to the Maximin criterion: MAMI and MADI. In this section the ability levels and the relative information function values (θ_k, r_k) are (-1.6, 2.0), (-.8, 5.4), (.0, 12.1), (.8, 21.3), (1.6, 10.8), and (2.4, 3.1), respectively.

The test information function values are shown in Table 3 for the method MAMI for values of P, 2 through 6.

Insert Table 3 here

The test information functions for P = 6 are displayed in Figure 3.

Insert Figure 3 here

For all values of P, ranging from 2 to 6 the information function of the last test constructed was considerably worse than the information function of the other tests. The aberrant information function in Figure 3 is the information function of the last test.

Table 4 shows the same results as Table 3, but now for MADI.

Insert Table 4 here

Figure 4 displays the information functions for P = 6.

Insert Figure 4 here

Comparing Figures 3 and 4, it can be seen that MADI does not suffer the drawback that later constructed tests tend to be worse than earlier constructed tests. This is because the P tests are constructed simultaneously.

Discussion

In this paper the methods MI and MIDI were proposed for constructing tests with a prespecified test information function. Similar methods namely MAMI and MADI were given for the construction of weakly parallel test approximately equal with respect to the Maximin criterion. The four methods were applied on a real item bank. In the next two sections the practicality of the methods is discussed.

Methods MI and MIDI

The methods MI and MIDI can be used for the construction of weakly parallel tests with a fixed target test information function. Both methods can construct tests in a couple of minutes from a bank of 600 items. Thus, CPU time is no obstacle in using the methods. The method MI has the drawback that tests are selected sequentially, so that tests constructed later tend to be worse than earlier constructed tests. In Table 1 it is seen that this problem occurs for P = 6. The information function of the sixth test is not as good as the one of the other five tests in the closeness of the match of test information function and target information function For the MIDI method the test information functions are closer to the target test information functic than for the MI method, because the tests are constructed simultaneously. MIDI is also faster than MI. In MIDI the heuristic DIFMIN is used and this heuristic is described here only for the case of practical constraints that imply a partition of the item bank. The drawback of MIDI is that the method is not applicable for other types of practical constraints.

Methods MAMI and MADI

The methods MAMI and MADI can be used to construct of weakly parallel tests with test information functions (approximately) optimal with respect to the Maximin criterion. If these methods are compared with respect to CPU time, it is clear that MADI is faster than MAMI. However, both methods can construct weakly parallel tests in a few minutes. The last test constructed by MAMI has always by far the worst test information function, because the "good" items have already been selected. In this respect MAMI is not as good as MADI. Method MADI, however, suffers the same drawback as MIDI.

References

Ackerman, T.A. (1989, March). An alternative methodology for creating parallel tests 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.

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 and M.R. Novick, Statistical theories of mental test scores. 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 (Dissertation). De Lier, The Netherlands: Academisch Boeken Centrum.

Boekkooi-Timminga, E. (1990). Parallel test construction from IRT-based item banks. Journal of Educational Statistics, 15.

Bruno, J.L., & Downey, P.J. (1986). Probabilistic bounds on the performance of list scheduling. SIAM J. Compt., 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.

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

Acknowledgement

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

Table 1

Information Functions for Weakly Parallel Tests Constructed by Method MI

P	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	1.936	5.285	12.073	21.073	10.991	2.971
2	1.805	5.256	12.218	21.430	10.841	2.902
3	2.037	5.788	12.456	21.568	10.389	3.072
4	2.051	5.368	12.061	21.166	10.674	3.033
5	1.898	5.371	12.278	21.446	10.717	2.853
6	2.517	4.779	13.029	20.406	11.271	3.236

Note. Total CPU-time: 1.97 minutes.

Table 2

Information Functions for Weakly Parallel Tests Constructed by Method MIDI

P	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	CPU (min)
2	1.976	5.137	11.992	21.530	10.755	3.122	0.55
	1.979	5.397	12.024	21.457	10.796	3.144	
3	2.099	5.583	12.281	21.518	10.742	3.054	0.63
	1.958	5.588	12.043	21.313	10.767	2.929	
	1.945	5.438	12.094	21.549	10.859	2.990	
4	2.030	5.480	11.980	21.314	10.693	2.962	0.61
	2.149	5.328	11.987	21.357	10.959	3.130	
	1.823	5.371	12.255	21.367	10.738	3.184	
	2.086	5.511	12.174	21.342	10.737	3.093	
5	2.080	5.351	12.169	21.447	10.716	3.106	0 63
	1.938	5.528	12.093	21.248	10.757	2.963	
	1.976	5.352	12.122	21.266	10.874	3.235	
	1.986	5.339	12.024	21.345	10.[illegible]53	3.100	
	1.968	5.327	11.986	21.371	10.833	3.033	
6	1.943	5.497	12.107	21.235	10.772	3.102	0.86
	2.146	5.507	12.083	21.156	10.709	3.101	
	2.180	5.364	12.049	21.590	10.653	3.518	
	1.868	5.365	12.070	21.215	11.075	3.091	
	2.074	5.382	12.072	21.200	10.817	3.073	
	1.912	5.613	12.516	21.308	10.563	3.045	

Table 3

Information Functions for Weakly Parallel Tests Constructed by Method MAMI

P	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	CPU (min)
2	3.078	8.722	18.369	31.670	16.529	4.689	1.50
	2.511	8.210	18.410	31.477	16.118	4.877	
3	2.818	7.790	17.241	30.079	15.734	4.714	1.94
	2.847	8.085	16.976	30.364	15.892	4.345	
	1.889	6.879	16.469	29.144	15.044	4.458	
4	2.496	7.293	16.318	28.241	14.976	4.433	2.16
	2.609	7.028	16.135	28.128	15.165	4.508	
	2.832	7.266	16.119	28.011	14.949	4.068	
	0.859	5.904	17.034	26.648	16.219	4.044	
5	2.600	7.260	15.253	26.699	14.326	3.669	2.58
	2.551	6.862	15.732	26.583	14.109	4.244	
	2.544	7.060	15.521	26.479	13.971	4.477	
	2.527	7.086	15.511	26.942	14.811	3.692	
	0.657	5.264	17.482	24.353	15.923	4.032	
6	2.163	6.795	15.985	24.751	14.133	3.923	2.73
	2.480	7.133	15.381	25.713	13.753	3.159	
	2.505	7.586	15.175	25.451	14.365	3.280	
	2.429	6.969	15.586	25.066	14.080	4.001	
	2.189	6.849	15.466	25.398	14.283	3.517	
	0.440	4.268	17.616	22.458	13.198	4.134	

Table 4

Information Functions for Weakly Parallel Tests Constructed by Method MADI

P	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	CPU (min)
2	3.107	8.727	18.184	31.873	16.583	4.955	0.63
	2.881	8.838	18.417	32.013	15.586	5.050	
3	2.792	7.835	17.276	30.388	15.453	4.624	0.61
	2.723	7.881	17.155	29.942	15.834	4.397	
	2.955	7.902	16.900	29.987	15.451	4.470	
4	2.909	7.467	16.308	28.193	14.587	4.186	0.65
	2.476	7.384	16.661	28.875	14.598	4.444	
	2.510	7.245	15.685	28.020	15.620	3.820	
	2.705	7.111	15.897	27.973	14.575	4.715	
5	2.532	6.783	15.920	27.515	14.149	4.657	0.65
	2.527	7.164	15.607	26.460	14.399	3.618	
	2.826	7.534	15.556	26.473	14.492	4.175	
	2.280	7.177	15.625	26.482	14.343	3.580	
	2.504	6.913	15.256	26.474	15.174	3.600	
6	2.262	6.863	15.790	25.343	13.869	3.544	0.70
	2.406	7.211	15.450	25.173	14.521	3.548	
	2.665	7.247	16.195	25.751	13.699	4.393	
	2.352	7.130	15.288	24.977	14.538	3.544	
	2.248	7.234	15.998	25.204	14.402	3.545	
	2.389	6.697	15.436	25.342	13.688	3.543	

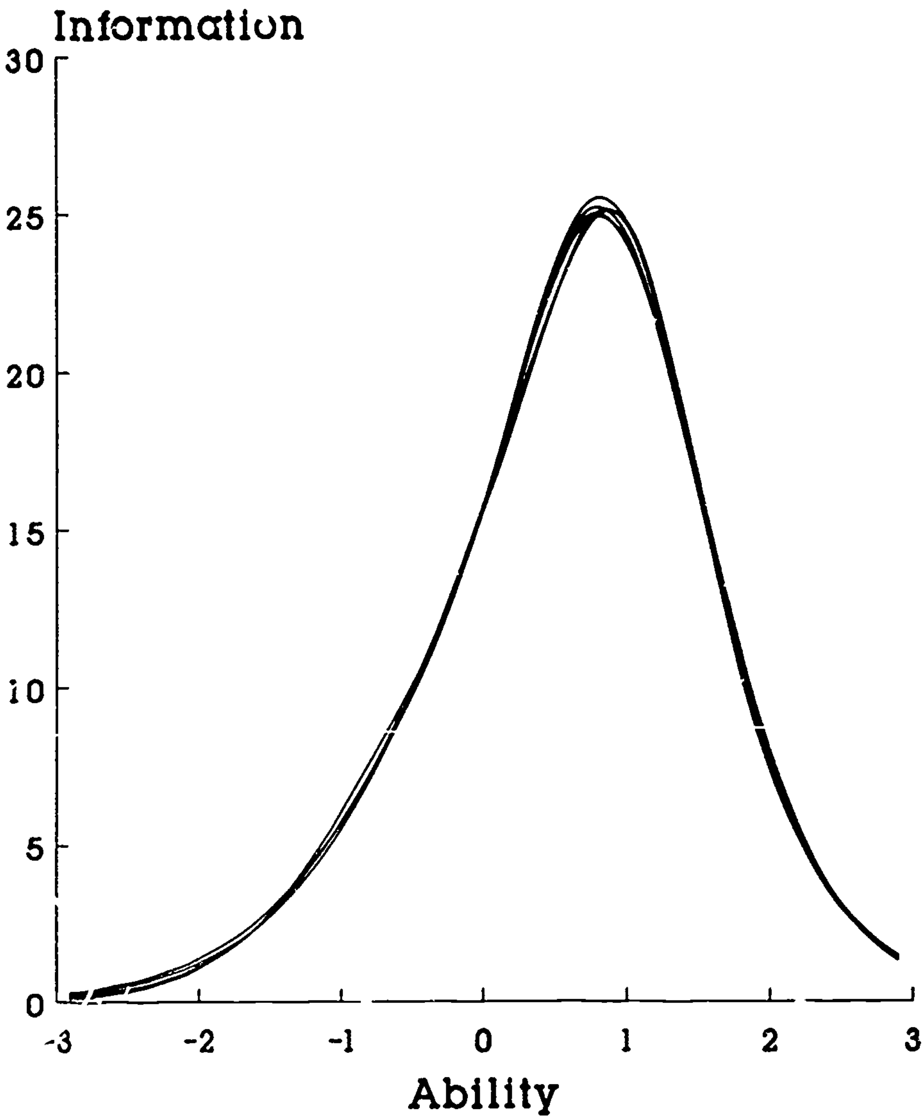
Figure Captions

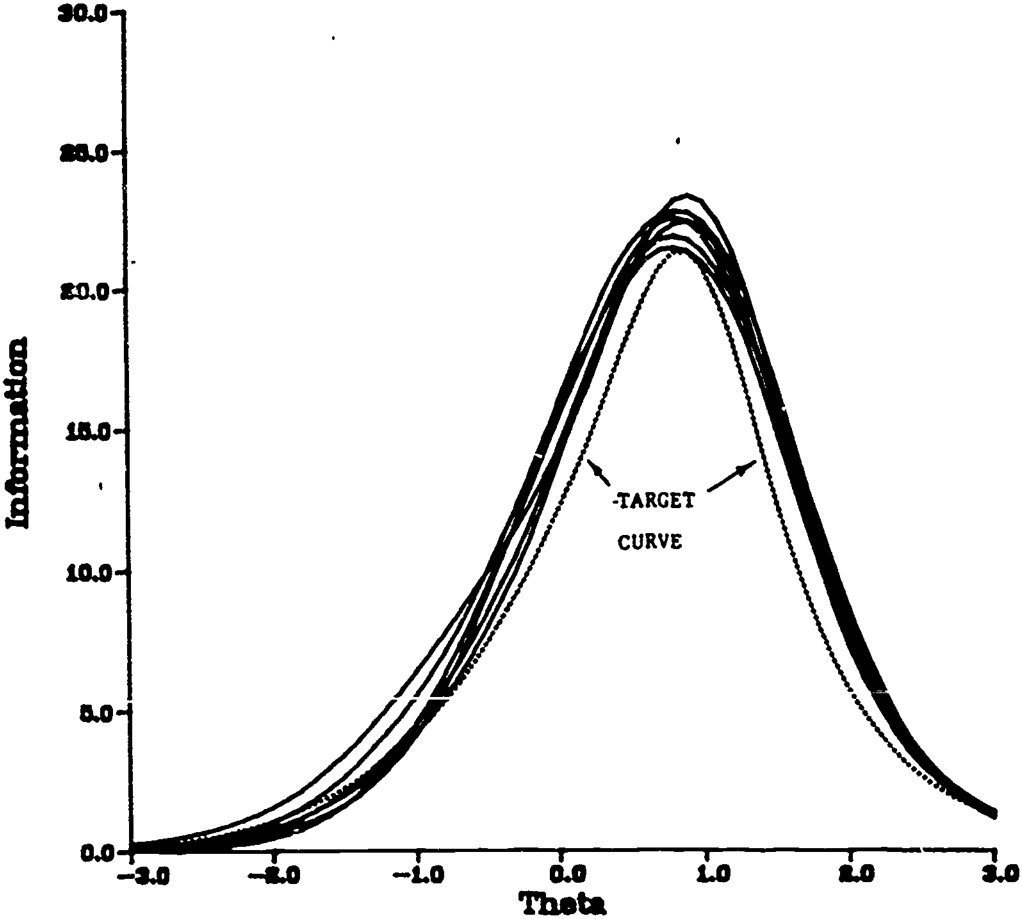
Figure 1. Information functions of tests constructed by method MI for P = 6.

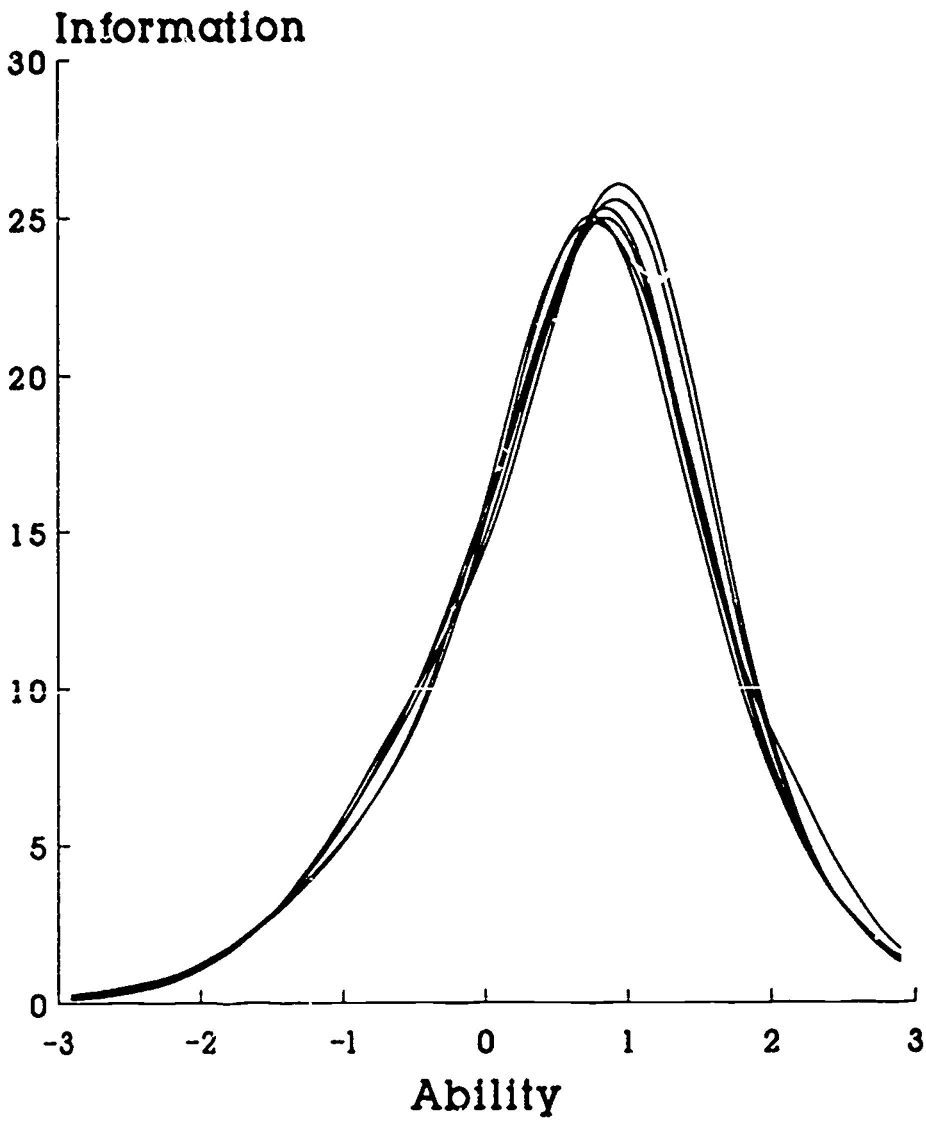
Figure 2. Information functions of tests constructed by method MIDI for P = 6.

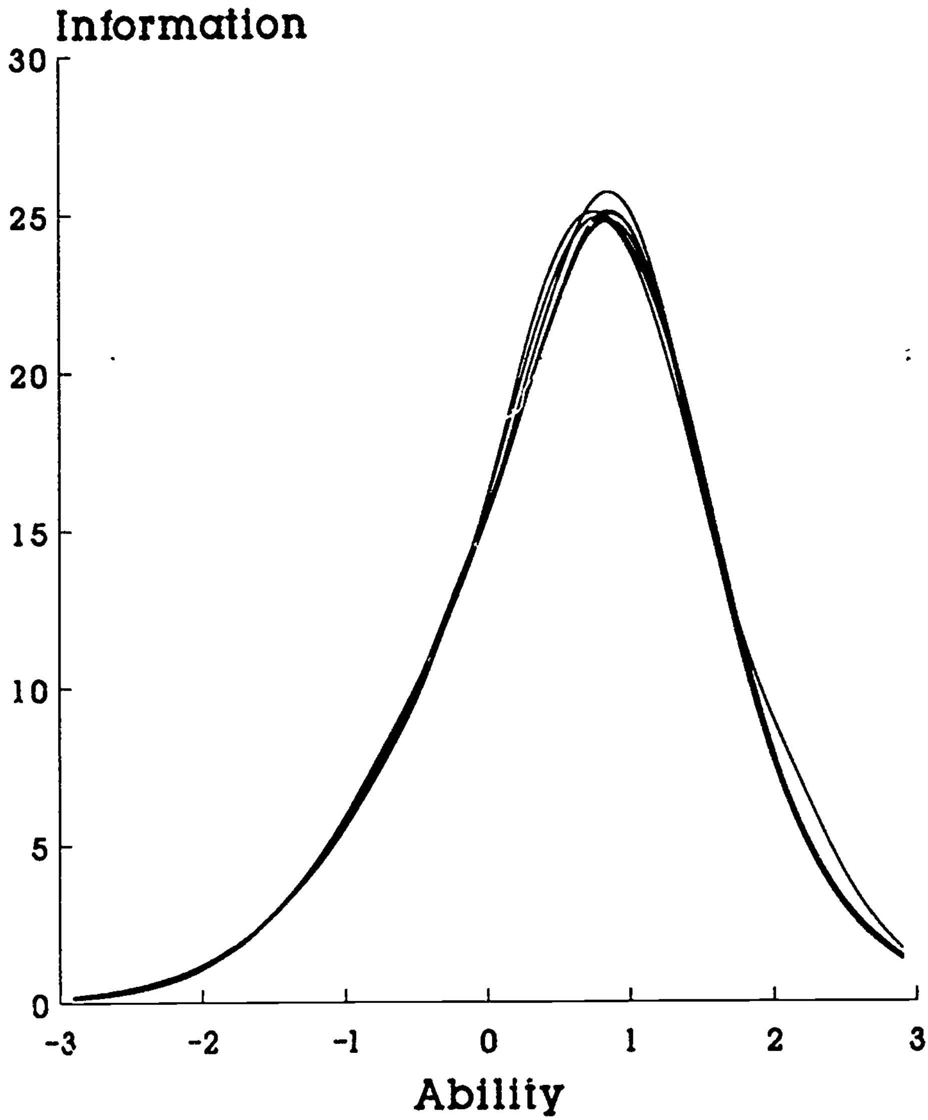
Figure 3. Information functions of tests constructed by method MAMI for P = 6.

Figure 4. Information functions of tests constructed by method MADI for P = 6.


Information
30
25
20
15
10
5
0
-3
-2
-1
0
1
2
3
Ability


Information
30.0
25.0
20.0
15.0
10.0
5.0
0.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0
Theta
TARGET
CURVE


Information
30
25
20
15
10
5
0
-3
-2
-1
0
1
2
3
Ability



Information
30
25
20
15
10
5
0
-3
-2
-1
0
1
2
3
Ability

Titles of recent Research Reports from the Division of Educational Measurement and Data Analysis, University of Twente, Enschede, The Netherlands.

RR-90-4 J.J. Adema, *Methods and Models for the Construction of Weakly Parallel Tests*

RR-90-3 H.J. Vos, *Simultaneous Optimization of Classification Decisions Followed by an End-of-Treatment Test*

RR-90-2 H. Tobi, *Item Response Theory at subject- and group-level*

RR-90-1 P. Westers & H. Kelderman, *Differential item functioning in multiple choice items*

RR-89-6 J.J. Adema, *Implementations of the Branch-and-Bound method for test construction problems*

RR-89-5 H.J. Vos, *A simultaneous approach to optimizing treatment assignments with mastery scores*

RR-89-4 M.P.F. Berger, *On the efficiency of IRT models when applied to different sampling designs*

RR-89-3 D.L. Knol, *Stepwise item selection procedures for Rasch scales using quasi-loglinear models*

RR-89-2 E. Boekkooi-Timminga, *The construction of parallel tests from IRT-based item banks*

RR-89-1 R.J.H. Engelen & R.J. Jannarone, *A connection between item/subtest regression and the Rasch model*

RR-88-18 H.J. Vos, *Applications of decision theory to computer based adaptive instructional systems*

RR-88-17 H. Kelderman, *Loglinear multidimensional IRT models for polytomously scored items*

RR-88-16 H. Kelderman, *An IRT model for item responses that are subject to omission and/or intrusion errors*

RR-88-15 H.J. Vos, *Simultaneous optimization of decisions using a linear utility function*

RR-88-14 J.J. Adema, *The construction of two-stage tests*

RR-88-13 J. Kogut, *Asymptotic distribution of an IRT person fit index*

RR-88-12 E. van der Burg & G. Dijksterhuis, *Nonlinear canonical correlation analysis of multiway data*

RR-88-11	D.L. Knol & M.P.F. Berger, *Empirical comparison between factor analysis and item response models*
RR-88-10	H. Kelderman & G. Macready, *Loglinear-latent-class models for detecting item bias*
RR-88-9	W.J. van der Linden & T.J.H.M. Eggen, *The Rasch model as a model for paired comparisons with an individual tie parameter*
RR-88-8	R.J.H. Engelen, W.J. van der Linden, & S.J. Oosterloo, *Item information in the Rasch model*
RR-88-7	J.H.A.N. Rikers, *Towards an authoring system for item construction*
RR-88-6	H.J. Vos, *The use of decision theory in the Minnesota Adaptive Instructional System*
RR-88-5	W.J. van der Linden, *Optimizing incomplete sample designs for item response model parameters*
RR-88-4	J.J. Adema, *A note on solving large-scale zero-one programming problems*
RR-88-3	E. Boekkooi-Timminga, *A cluster-based method for test construction*
RR-88-2	W.J. van der Linden & J.J. Adema, *Algorithmic test design using classical item parameters*
RR-88-1	E. van der Burg & J. de Leeuw, *Nonlinear redundancy analysis*

Research Reports can be obtained at costs from Bibliotheek, Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

...tment of

...UCATION

A publication by
the Department of Education
of the University of Twente

...de
...nd