ED 310 120   TM 013 693

AUTHOR   Boekkooi-Timminga, Ellen
TITLE   Simultaneous Test Construction by Zero-One
                Programming. Research Report 86-3.
INSTITUTION   Twente Univ., Enschede (Netherlands). Dept. of
                Education.
SPONS AGENCY   Netherlands Inst. for the Advancement of Pure
                Research, The Hague.
PUB DATE   86
NOTE   30p.; Also cited as Project Psychometric Aspects of
                Item Banking No. 4. In "Methodika," 1986, 1.
PUB TYPE   Reports - Evaluative/Feasibility (142)

EDRS PRICE   MF01/PC02 Plus Postage.
DESCRIPTORS   *Item Banks; *Latent Trait Theory; *Mathematical
                Models; Operations Research; *Test Construction; Test
                Items
IDENTIFIERS   *Parallel Test Forms; Simultaneous Processing; *Zero
                One Programing

ABSTRACT
        A method is described for simultaneous test
construction using the Operations Research technique zero-one
programming. The model for zero-one programming consists of two
parts. The first contains the objective function that describes the
aspect to be optimized. The second part contains the constraints
under which the objective function should be optimized. The selection
of items is based on information from item response theory.
Simultaneous test design is used when tests have to be constructed so
that there is a certain relationship between them. Two examples of
simultaneous test construction are presented. The construction of two
parallel tests is considered, and designs of three tests that should
measure best at successive parts of the ability scale are described.
Examples were carried out using an item bank of 10 items chosen at
random. The sequential construction of the same series of tests was
compared. The examples illustrate that the tests constructed using
simultaneous techniques best fit the intentions of the test
constructor. Three tables give the data for the test construction
methods. (Author/SLD)

Simultaneous Test Construction by Zero-One Programming

E. Boekkooi-Timminga

Division of
Educational Measurement
and Data Analysis

BEST COPY AVAILABLE

Department of Education

University
of
Twente

Project Psychometric Aspects of Item Banking No. 4

\

3

Simultaneous Test Construction

by Zero-One Programming

Ellen Boekkooi-Timminga

Abstract

A method for simultaneous test construction, using the
Operations Research technique zero-one programming is described.
The selection of items is based on the concept of information from
item response theory.

Simultaneous test design is to be used when tests have to be
constructed that have a certain relationship between them. Some
examples that conclude the paper show the advantage of using
simultaneous tecnniques over methods that construct tests one after
another.

1. Introduction

One of the main applications of item banking is test construction. Recently, a start has been made with research on systematic test construction based on operations research techniques (Boekkooi-Timminga, 1986; Theunissen, 1985; Theunissen & Verstralen, 1986; Timminga, 1985; van der Linden & Boekkooi-Timminga, 1986). The methods developed so far are all methods for the construction of one test at a time.

For many test design problems these methods are applicable; for some, however, they will not be appropriate. This is the case when the tests to be constructed have a certain relationship to each other, for instance, parallel tests or two-stage testing procedures (Lord, 1980). A possible relation is that the tests have no items in common, or that they nave increasing difficulties or test lengths. Of course, such tests can be constructed one at a time, but this has some disadvantages. When the first test is constructed, the other tests to be constructed can not be taken into account. A consequence is that an item can be selected for the first test while it would fit one of the other tests much better. This disadvantage becomes greater the larger the number of tests to be designed.

Following Birnbaum (1968), as an instrument for test construction the concept of information from item response theory is used. The information function for an unbiased consistent estimator of abi'ity is defined as the reciprocal of the asymptotic

sampling variance of the estimator. For the three-parameter logistic model, which considers only dichotomous items, the item information function can be expressed as

$$(1) \qquad I_i(\theta) = \frac{a_i^2 (1-c_i)}{(c_i + \exp(a_i(\theta - b_i)))(1 + \exp(-a_i(\theta - b_i)))^2} \quad ,$$

where a_i, b_i, c_i are the discrimination, difficulty and guessing parameters for item i and θ is the ability level considered. The relation between the test $I(\theta)$ and item information functions $I_i(\theta)$ is formulated as

$$(2) \qquad I(\theta) = \sum_{i=1}^{n} I_i(\theta) \quad ,$$

where $I(\theta)$ is an upper bound for any scoring method.

Test construction in this article and the publications mentioned above is based on the specification of a target test information function. Those items are selected for which the actual test information function approximates the desired function best. Pecause it is in genera: not interesting to consider the whole information function, only the values corresponding to some specific points at the ability scale are considered. Furthermore, it is assumed that the items in the item bank considered are all calibrated and fit the same one-dimensional model.

In this paper a model for simultaneous test construction based on zero-one programming is described. Then, the results of two

examples are given in which two parallel tests and three tests, which measure best at successive parts of the ability scale, are designed in two different manners: simultaneously and one by one. The paper ends with some conclusions and remarks.

2. A Model for Simultaneous Test Construction

The model for simultaneous test construction is based on the Operations Research technique zero-one programming. Some references to these techniques are: D-ellenbach, George, and McNickle (1983), Syslo, Deo, and Kowalik (1983), Taha (1975), and Wagner (1972). A model for zero-one programming exists of two parts. The first part contains the objective function which describes the aspect to be optimized. The second part of the model contains the constraints under which the objective function should be optimized. A restriction to both objective functions and constraints is that they all have to be linear.

In this section first the symbols used in the test construction model are given. Next, a number of possible constraints are formulated. The section ends with some useful objective functions in simultaneous test design.

2.1. Definitions of Symbols

Constants

i : items in the bank $(i = 1,...,N)$

n : total number of items to be selected

n_t : number of items to be selected for test t

t : tests to be constructed $(t = 1,...,T)$

k : points on the ability scale to be considered $(k = 1,...,K)$

I_{ik} : information of item i at ability level k

D_{tk} : target test information for test t at ability level k

Variables

$$x_{it} : \text{decision variable} \begin{cases} 1 & \text{item i is selected for test t} \\ \\ 0 & \text{otherwise} \end{cases}$$

2.2. Some Constraints

In test design many constraints may apply. Some of the most frequently occuring constraints in simultaneous test design will be mentioned here. For a more thorough description of possible constraints in test design the reader is referred to van der Linden and Boekkooi-Timminga (1986). In this reference constraints are given for the case that one test at a time is constructed. These are however easily generalized for the case of simultaneous test design, by adjusting the decision variables.

The first constraints to be mentioned involve the range of values of the decision variables. They may only take the values zero and one, indicating for each test if item i must be part of it or not.

(3) $x_{it} \in \{0,1\}$ $i = 1,\ldots,N$
$t = 1,\ldots,T$

The specification of the target information functions for the tests to be constructed requires a constraint for each test at each of the ability levels considered. The desired test information for test t at ability level k is given by

(4) $\sum_{i=1}^{N} x_{it} I_{ik} \overset{>}{\underset{<}{}} D_{tk}$

The sign in this formula will be "greater than" in most instances, because generally it is preferred getting more information than desired than less.

It is optional to put constraints on the number of items to be selected. Only in some instances the number of items should not be fixed. This is the case when the number of items is part of the objective fuction, for instance, when the number of items to be selected has to be minimized. For each test the exact number or an upper or lower limit on the number of items in the test can be specified. For test 1 this is formulated as

(5) $\sum_{i=1}^{N} x_{i1} \overset{>}{\underset{<}{=}} n_t$

It is also possible to put restrictions on the total number of items for all tests to be selected

(6) $$\sum_{i=1}^{N} \sum_{t=1}^{T} x_{it} \overset{>}{\underset{<}{=}} n$$

Further, it is possible to put constraints on the ratio of the number of items selected for each test, described by $a_1, a_2, \ldots, a_T$

(7) $$a_1 \sum_{i=1}^{N} x_{i1} = a_2 \sum_{i=1}^{N} x_{i2} = \ldots \ldots = a_T \sum_{i=1}^{N} x_{iT}$$

Besides putting constraints on the number of items to be selected for the tests, it is also possible to influence the individual selection of items. A simple way to include or exclude some specific items is to fix the decision variables of these items to one or zero, for instance, excluding item 2 from all tests

(8) $$x_{2t} = 0 \qquad t = 1, \ldots, T \ ,$$

or including item 2 in one of the T tests

(9) $$\sum_{t=1}^{T} x_{2t} = 1 \ ,$$

or including all items in maximal one test

(10) $$\sum_{t=1}^{T} x_{it} < 1 \qquad i = 1, \ldots, N$$

In the same manner it is also possible to specify for some of the tests that they may have no items in common.

2.3. Objective Functions

In this paragraph some objective functions are given that can be used in simultaneous test construction. First, some functions are described that are also to be used in the construction of one test at the time. Then, a few objective functions are mentioned to be used for simultaneous test design only.

The first objective function that was introduced in test construction by zero-one programming involved the number of items to be selected (Theunissen, 1985). Theunissen considered the construction of one test at a time as a multi-dimensional knapsack problem (e.g., Taha, 1975). He minimized the number of items to be selected subject to the constraints that the actual test information at some specified ability levels should exceed certain predetermined heights. In simultaneous test construction this objective function is also applicable. The total number of items to be selected for all tests or one of the tests is then minimized. When the total number of items to be selected for all tests to be constructed is minimized, the objective function looks like

$$(11) \qquad \text{minimize} \quad \sum_{i=1}^{N} \sum_{t=1}^{T} x_{it}$$

Objective functions involving the number of items to be selected have both an important advantage and disadvantage. The advantage is that with a minimum number of items a maximum amount of information can be obtained. The disadvantage concerns the practical

applicability of the function; many test constructors may want to have the opportunity to influence the number of items to be selected.

An objective function that does not have this disadvantage involves minimizing the distance between the target and the actual test information function. This objective function in test construction was first looked at by Timminga (1985). It was formulated for the construction of one test as minimizing the sum of the distances between the target and actual test information function over all ability levels considered. This is equivalent to minimizing the sum of the actual test information over all ability levels, under the constraints that for each ability level the actual test information exceeds the value of the target function D_k. For each item a decision variable x_i was introduced indicating for item i if it must be included in the test or not. This objective function is formulated as follows

(12) minimize $\sum_{i=1}^{N} \sum_{k=1}^{K} x_i I_{ik}$

subject to

(13) $\sum_{i=1}^{N} x_i I_{ik} > D_k$ $k = 1,\ldots,K$

Another objective function is to maximize this sum under the restriction that the actual test information should not exceed the target function value. This function is not looked at any further

here. Formulated for a simultaneous test design model, (12) and
(13) are transformed into

(14)   minimize $\sum\limits_{i=1}^{N} \sum\limits_{k=1}^{K} \sum\limits_{t=1}^{T} x_{it} I_{ik}$

          subject to

(15) $\sum\limits_{i=1}^{N} x_{it} I_{ik} > D_{tk}$ $t = 1,\ldots,T$
                                                            $k = 1,\ldots,K,$

where the actual test information of all tests is summed in (14).
In (15) for each test at each ability level considered a minimum
height of the test information to be obtained is given.

Objective functions are viewed more intensively in van der
Linden and Boekkooi-Timminga (1986), where the construction of one
test at a time is considered. The objective functions given there
can be easily generalized to the case that more than one test has
to be constructed by adjusting the decision variables. The same was
done for instance by transforming the objective function (12) -
(13) into (14) - (15).

Besides objective functions that are used in constructing one
test at · time and that can be generalized for the construction of
more tests at a time, there are also functions to be used
exclusively in simultaneous test design. These are objective
functions that involve the actual test information functions of the
tests constructed. For instance, when parallel tests have to be

constructed, minimizing the sum of the absolute distances between the actual test information functions. Also, functions like minimizing the maximum absolute distance between the actual test information functions for the tests constructed are possible. For the construction of two tests at the same time this is formulated as

(16)   minimize y

         subject to

(17)
$$\sum_{i=1}^{N} x_{i1} I_{ik} - y < \sum_{i=1}^{N} x_{i2} I_{ik} \qquad k = 1,\ldots,K$$

(18)
$$-\sum_{i=1}^{N} x_{i1} I_{ik} - y < -\sum_{i=1}^{N} x_{i2} I_{ik} \qquad k = 1,\ldots,K \ ,$$

where y is the maximum distance between the two realized test information functions. The ultimate choice of an objective function will depend on the specific test design problem and the wishes of the test constructor.

3. Some Examples

In this section two examples of simultaneous test construction are presented. First, the construction of two parallel tests is considered, and second, the design of three tests, which measure

best at successive parts of the ability scale is described. In both cases the tests are constructed simultaneously as well as one test at a time.

The examples were carried out using a fictive item bank containing ten items chosen at random. Only ten items were considered, because of the complexity of solving zero-one problems with larger numbers of variables and constraints (Theunissen, 1985; Taha, 1975; Van der Linden & Boekkooi-Timminga, 1986).

INSERT TABLE 1 ABOUT HERE

In Table 1 the item parameters and the item information at three ability levels (θ = -1, 0, 1) for the items in the item bank are given. For each item it is assumed that the guessing parameter is equal to 0. The item difficulties vary from -1.12 to 0.92; the item discrimination indices have values between 0.57 and 1.60. The computer program for zero-one programming that was used was based on an implementation on a DEC-2060 mainframe of the algorithm of Land and Doig (1960), which is a branch-and-bound method.

The first example concerned the design of two parallel tests. Tests are considered parallel if they have the same test information functions (Samejima, 1977). From a practical point of view it was desired that the tests should contain an equal number of items. The actual test information for both tests at the three

ability levels considered was desired to be at least 0.50. The
tests were designed simultaneously twice using different objective
functions. First, as an objective function the sum of the distances
between the target and actual test information functions at the
specified ability levels was considered. The other objective
function minimized the maximum absolute distance between the actual
test information functions of both parallel tests. The simultaneous
test construction model for the first case was

$$\text{(19)} \qquad \text{minimize} \quad \sum_{i=1}^{10} \sum_{t=1}^{2} \sum_{k=1}^{3} x_{it} I_{ik}$$

subject to

$$\text{(20)} \qquad \sum_{i=1}^{10} x_{it} I_{ik} > 0.5 \qquad \begin{array}{l} t = 1,2 \\ k = 1,2,3 \end{array}$$

$$\text{(21)} \qquad \sum_{i=1}^{10} x_{i1} = \sum_{i=1}^{10} x_{i2}$$

$$\text{(22)} \qquad x_{i1} + x_{i2} < 1 \qquad i = 1,\ldots,10$$

$$\text{(23)} \qquad x_{it} \, \epsilon \, \{0,1\} \qquad \begin{array}{l} i = 1,\ldots,10 \\ t = 1,2 \end{array}$$

In the case of minimizing the maximum absolute distance (19) was
replaced by (16) and the constraints (17) and (18) had to be added
to the test construction model mentioned above. When the tests were
constructed one at a time, the following test design model for the

first test was used

$$(24) \qquad \text{minimize} \quad \sum_{i=1}^{10} \sum_{k=1}^{3} x_i I_{ik}$$

subject to

$$(25) \qquad \sum_{i=1}^{10} x_i I_{ik} > 0.5 \qquad\qquad k = 1,2,3$$

$$(26) \qquad x_i \in \{0,1\} \qquad\qquad i = 1,\ldots,10$$

The test design model for the second test was the same, except for constraints that had to be added to exclude the items selected for the first test, and to determine that the number of items to be selected should be the same as for the first test.

INSERT TABLE 2 ABOUT HERE

The results of this example considering the design of parallel tests are summarized in Table 2. A remarkable difference between the two types of test design was noticed. In constructing the tests sequentially there was a great discrepancy between the actual test information functions of both parallel tests constructed which did not occur when the tests were designed simultaneously. Furthermore, it was seen that most items were selected for an individual test in

sequential test design. Looking at the items selected it was remarked that in the case of constructing the two tests one after another those items were selected for the first test which had very low item information values at the ability levels considered. This has to be explained from the objective function used minimizing the distance between the target and actual test information function. Thus, more items had to be selected and the actual test information functions differed more from each other than in simultaneously designed tests. In comparing the two simultaneous test constructing methods it was seen that the differences between the actual test information functions were smaller when the maximum absolute distance between both functions was minimized. This was to be expected since the actual test information functions were indirectly connected to each other by the target test information function in the objective function. The problem of designing parallel tests is considered more explicitly in Boekkooi-Timminga (1986).

The second example carried out involved the design of three tests which should measure best at successive parts of the ability scale. For the tests it was desired that the actual test information should exceed 0.75 at the ability levels $\theta = -1, 0, 1$, respectively. Further, it was demanded that the tests had no items in common. The objective function used was minimizing the total number of items to be selected for the tests to be constructed. The model formulation for simultaneously designing these tests was

(27)   minimize $\displaystyle\sum_{i=1}^{10} \sum_{t=1}^{3} x_{it}$

subject to

(28) $\displaystyle\sum_{i=1}^{10} x_{i1} I_{i1} > 0.75$

(29) $\displaystyle\sum_{i=1}^{10} x_{i2} I_{i2} > 0.75$

(30) $\displaystyle\sum_{i=1}^{10} x_{i3} I_{i3} > 0.75$

(31) $x_{i1} + x_{i2} + x_{i3} < 1$ $i = 1,\ldots,10$

(32) $x_{it} \in \{0,1\}$ $i = 1,\ldots,10$
                                        $t = 1,2,3$

When the tests were constructed one at a time, the test construction model for the first test which measured best at ability level $\theta = -1$, was

(33)   minimize $\displaystyle\sum_{i=1}^{10} x_i$

subject to

(34)
$$\sum_{i=1}^{10} x_i I_{i1} > 0.75$$

(35)
$$x_i \in \{0,1\} \qquad\qquad i = 1,\ldots,10$$

The test design models for the second and third test were similar.
The only difference was that the ability level considered changed,
and that extra constraints had to be added to exclude the items
that were already selected for the other test(s).

INSERT TABLE 3 ABOUT HERE

The results of the construction of the three tests are described in
Table 3. It was seen that in the case of sequential test
construction no solution was found when the third test was to be
constructed. The test information for the items not yet selected
was 0.718, which was below the target value of 0.75. The results of
the simultaneous design, however, showed that it was possible to
construct all three tests desired. This example showed clearly the
advantage of simultaneous test design over sequential test design,
which does not take the other test(s) to be constructed into
account. Remarkably, for both methods the actual test information
function of the first test looked a little strange, because the
amount of test information obtained at the second ability level was
higher than that at the first level. This can be explained from the

properties of the items in the item bank considered, and from the test construction model in which it is not explicitely determined that the amount of test information for each test must be highest at the ability level considered.

4. Discussion

In this paper it is shown how tests can be constructed simultaneously using zero-one programming. Simultaneous test design is to be used when tests have to be constructed that should have a certain relationship to each other. From the examples given, in which tests were constructed both simultaneously and sequentially, it appeared that the tests designed simultaneously corresponded most to the desires the test constructor had. Especially in the first example, designing the tests one after another, it is noticed that items are selected for the first test that should fit the second test to be constructed better. This problem does not occur when the tests are constructed simultaneously. Of course, some of the deviations can be partly explained from the small item bank used and the specific objective function that was minimized, but these aspects influence both types of test construction.

References

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. London: Addison-Wesley Publishing Company, Inc.

Boekkooi-Timminga, E. (1986) Algorithms for the construction of parallel tests by zero-one programming. Enschede, The Netherlands: University of Twente.

Daellenbach, H.G., George, J.A., McNickle, D.C. (1983) Introduction to operations research techniques (2nd ed). London: Allyn and Bacon, Inc.

Land, A.H., Doig, A. (1960) An automatic method of solving discrete programming problems. Econometrica, 28, 497-520.

Lord, F.M. (1980) Applications of item response theory to practical testing problems. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers.

Samejima, F. (1977) A use of the information function in tailored testing. Applied psychological measurement, 1, 233-247.

Syslo, M.J., Deo, N., Kowalik, J.S. (1983) Discrete optimization algorithms. Englewood Cliffs, New Jersey: Prentice-Hall.

Taha, H.A. (1975) Integer programming. New York: Academic Press.

Theunissen, T.J.J.M. (1985) Binary programming and test design. Psychometrika, 50, 411-420.

Theunissen, T.J.J.M., Verstralen, H.H.F.M. (1986) Algoritmen voor
het samenstellen van toetsen Algorithms for test construction .
In W.J. van der Linden (Ed.), Moderne methoden voor
toetsgebruik en -constructie. Lisse, The Netherlands: Swets &
Zeitlinger.

Timminga, E. (1985) Geautomatiseerd toetsontwerp: itemselectie
m.b.v. binair programmeren. Automated test design: Item
selection by binary programming . Enschede, The Netherlands:
University of Twente.

van der Linden, W.J.. Boekkooi-Timminga, E. (1986) Algorithmic test
design with practical constraints. Enschede, The Netherlands:
University of Twente.

Wagner, H.M. (1972) Principles of operations research: with
applications to managerial decisions. Londen: Prentice-Hall
International, Inc.

Author's Note

Table 1

Item Parameters and Item Information at Three Ability Levels

Item	b	a	$I_i(-1)$	$I_i(0)$	$I_i(1)$
1	0.48	0.79	0.113	0.151	0.150
2	-0.29	1.60	0.471	0.607	0.256
3	0.54	1.41	0.183	0.432	0.448
4	0.92	0.98	0.110	0.197	0.240
5	0..	0.66	0.090	0.108	0.104
6	-1.12	0.78	0.152	0.126	0.082
7	-0.50	0.59	0.085	0.085	0.072
8	-0.52	1.57	0.536	0.524	0.190
9	0.12	1.46	0.291	0.529	0.362
10	-0.35	0.57	0.079	0.080	0.070

Table 2

Items Selected for Two Parallel Tests

           Items Selected   I(-1)   I(0)   I(1)

Simultaneous Construction

Minimizing the sum of distances between target and obtained test
information functions

Test 1	6 - 9 - 10	0.522	0.735	0.514
Test 2	4 - 7 - 8	0.731	0.806	0.502

Minimizing the maximum absolute distance between both obtained test
information functions

Test 1	1 - 2 - 6 - 10	0.815	0.964	0.558
Test 2	4 - 5 - 7 - 8	0.821	0.914	0.606

Sequential Construction

Test 1	4 - 5 - 6 - 7 - 10	0.516	0.596	0.568
Test 2	3 - 8	0.719	0.956	0.638

Table 3

Item Selected for Three Tests which Measure Best at

Successive Parts of the Ability Scale

	Items Selected	I(-1)	I(0)	I(1)

Simultaneous Construction

Test 1	8 - 9	0.927	1.053	0.552
Test 2	1 - 2	0.584	0.758	0.406
Test 3	3 - 4 - 5	0.383	0.737	0.792

Sequential Construction

Test 1	2 - 8	1.007	1.131	0.446
Test 2	3 - 5	0.474	0.961	0.810
Test 3	infeasible			

Titles of Recent Research Reports

RR-86-1 W.J. van der Linden, The use of test scores for
classification decisions with threshold utility

RR-86-2 H. Kelderman, Item bias detection using the loglinear
Rasch model: Observed and unobserved subgroups

RR-86-3 E. Boekkooi-Timminga, Simultaneous test construction by
zero-one programming

RR-86-4 W.J. van der Linden, & E. Boekkooi-Timminga, A zero-one
programming approach to Gulliksen's random matched
subtests method

RR-86-5 E. van der Burg, J. de Leeuw, & R. Verdegaal, Homogeneity
analysis with k sets of variables: An altenating least
squares method with optimal scaling features

RR-86-6 W.J. van der Linden, & T.J.H.M. Eggen, An empirical Bayes
approach to item banking

RR-86-7 E. Boekkooi-Timminga, Algorithms for the construction of
parallel tests by zero-one programming

RR-86-8 T.J.H.M. Eggen, & W.J. van der Linden, The use of models
for paired comparisons with ties

A publication by
the Department of Education
of the University of Twente,
P.O. Box 217,
7500 AE Enschede,
The Netherlands

Department of Education

30