European Journal of Operational Research 55 (1991) 103–111
North-Holland

Theory and Methodology

Achievement test construction using 0–1 linear programming

Jos J. Adema, Ellen Boekkooi-Timminga and Wim J. van der Linden
Department of Education, University of Twente, P.O. Box 217, 7500 AE Enschede, Netherlands

Received February 1989; revised November 1989

Abstract: In educational testing the work of professional test agencies has shown a trend towards item banking. Achievement test construction is viewed as selecting items from a test item bank such that certain specifications are met. As the number of possible tests is large and practice usually imposes various constraints on the selection process, a mathematical programming approach is obvious. In this paper it is shown how to formulate achievement test construction as a 0–1 linear programming problem. A heuristic for solving the problem is proposed and two examples are given. It is concluded that a 0–1 linear programming approach fits the problem of test construction in an appropriate way and offers test agencies the possibility of computerizing their services.

Keywords: Item banking, achievement test construction, zero–one programming, heuristics

1. Introduction

Some thirty years ago psychometric theory began to use stochastic response models to objectively estimate such properties of achievement test items as their difficulty and discriminating power. In the 1980's this development has led to the notion of item banking. Test item banking assumes the existence of large collections of test items stored in a computer together with accurate estimates of their measurement properties. The ultimate aim is to use the information on the items for tailoring tests to educational specifications. Nowadays, many test agencies, like the Educational Testing Service (ETS) in the USA and the National Institute for Educational Measurement (CITO) in The Netherlands, are developing computerized systems for handling item banks. These systems include procedures for test construction, adaptive test administration, item parameter estimation, and diagnosing and scoring response vectors. A complete description of a computerized test system is given in van Thiel and Zwarts (1986).

From an item bank of a typical size (say 300–500 items) a large number of different tests can be constructed. Also, various constraints with respect to test content, item format, administration time, and history of previous item usage may have to be imposed on the final product. Therefore, a mathematical programming approach to the problem seems obvious. The idea to use mathematical programming was already suggested by Yen (1983). However, Theunissen (1985) was the first to formulate an optimization model for the problem. The use of mathematical programming was further explored in a series of papers by Boekkooi-Timminga (1987, 1990), Gademann (1987), Adema and van der Linden (1989), and van der Linden and Boekkooi-Timminga (1989).

It is the purpose of this paper to show how test construction can be formulated as a 0–1 linear programming problem and to summarize some of the results. Before proceeding, the notion of an information function from item response theory (IRT) will be introduced; information functions play a central role in the models proposed. Then, two 0–1 linear programming models for test construction, and a number of practical constraints are presented. Next, a heuristic used to solve large-scale applications is described. The paper is concluded with two realistic examples illustrating the possibilities of computerized achievement test construction.

2. Item response theory

In item response theory (e.g., Fischer, 1974; Hambleton and Swaminathan, 1985; Lord, 1980; Rasch, 1960) the probability of a correct response to a test item is modelled as a function of the ability of the examinee and certain characteristics of the item. Several kinds of IRT-models have been developed. The most popular models consider dichotomously scored (right–wrong) responses to items measuring a single ability. Generalizations to polytomous or graded response formats as well as models for vector-valued abilities are amply available. For the sake of illustration, a three-parameter logistic model for dichotomous responses is considered.

Let θ be a scalar representing the ability of the examinees on the items in the bank. Then, for item $i = 1, \ldots, I$ the probability of a correct response for an examinee with ability value θ is modelled as

$$P_i(\theta) = c_i + (1 - c_i) \times \{1 + \exp[-a_i(\theta - b_i)]\}^{-1}, \qquad (1)$$

where

$\theta \in \langle -\infty, +\infty \rangle$, $a_i \in [0, +\infty\rangle$,

$b_i \in \langle -\infty, +\infty \rangle$, and $c_i \in [0, 1]$

(Lord, 1980). The parameters b_i and a_i denote the difficulty and the discriminating power of item i, respectively, whereas c_i is the probability of solving item i correctly for $\theta \to -\infty$ (guessing parameter for multiple-choice items). $P_i(\theta)$ is called the item characteristic function. In Figure 1 an item characteristic function is given, and the interpretation of the item parameters is explained. Figure 2 gives the characteristic functions of three different items.

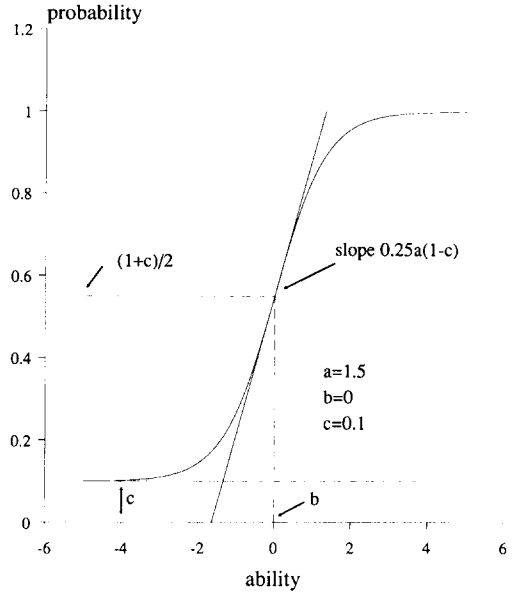

Figure 1. Item characteristic curve

A simplified, computationally very attractive model is the Rasch (1960) or one-parameter logistic model. It assumes that $c_i = 0$ and that a_i is equal for all items. Of course, the Rasch model

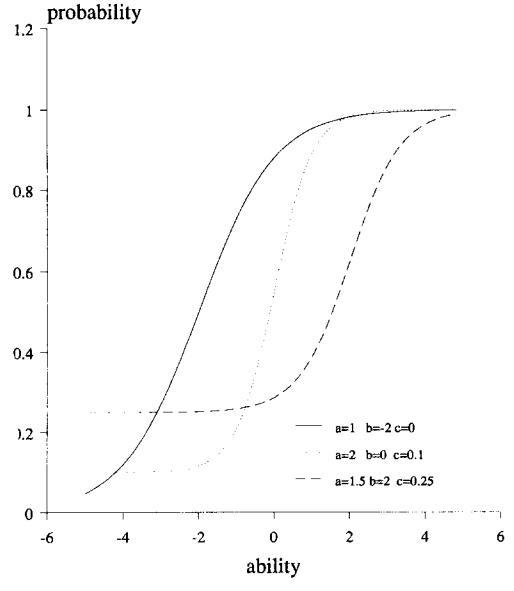

Figure 2. Three item characteristic curves

should only be used if the assumptions show a satisfactory fit to the test data at hand.

The usual procedure in item banking is to use response data from a sample of examinees to estimate the parameters of the items in the bank. Common estimation methods are maximum likelihood or Bayesian posterior modal estimation. Since it is impossible to administer all items to all examinees in the sample, the estimation problem is one with incomplete data and sample optimization is possible. For this purpose 0–1 linear programming methods can be used (van der Linden, 1988); a discussion of the results is beyond the scope of this paper. Once their parameters are known, the items can be used to estimate the ability of each new examinee. For reasons that will become clear immediately, maximum-likelihood estimation is used.

Let $U_1, \ldots, U_n$ denote the responses of a new examinee to an n-item test from the bank. Under local independence, that is, independence between the response variables for a fixed value of θ, the likelihood associated with a response vector $u_1, \ldots, u_n$ is equal to

$$L(\theta;\, u_1, \ldots, u_n, \mathbf{a}, \mathbf{b}, \mathbf{c},) = \prod_{i=1}^{n} P_i(\theta)^{u_i}\left[1 - P_i(\theta)\right]^{1-u_i},$$

where

$$\mathbf{a} = (a_1, \ldots, a_n), \quad \mathbf{b} = (b_1, \ldots, b_n), \quad \text{and}$$
$$\mathbf{c} = (c_1, \ldots, c_n).$$

Hence, the ability parameter θ can be estimated from the likelihood equation

$$\frac{\partial}{\partial\theta} \ln L(\theta;\, u_1, \ldots, u_n, \mathbf{a}, \mathbf{b}, \mathbf{c}) = \frac{\partial}{\partial\theta}\left(\sum_{i=1}^{n} u_i \ln\left[P_i(\theta)\right] + (1 - u_i) \ln\left[1 - P_i(\theta)\right] \right) = 0.$$

For the model in (1) the likelihood equation takes the form

$$\sum_{i=1}^{n} \left[u_i - P_i(\theta)\right] \frac{P_i'(\theta)}{P_i(\theta)\left[1 - P_i(\theta)\right]} = 0,$$

with $P_i'(\theta) = \partial/\partial\theta P_i(\theta)$, while for the Rasch model it reduces to

$$\sum_{i=1}^{n} \left[u_i - P_i(\theta)\right] = 0.$$

The likelihood functions can be solved, for instance, by the Newton–Raphson procedure.

It should be noted that, because of the presence of accurate estimates of the item parameters in the model, the likelihood equations correct the ability estimates for the properties of the items. Therefore, no matter the selection of the items made, all examinees are scored on the same scale. This is an attractive advantage over traditional test scoring where no implicit score equating is available.

A well-known measure for the information in a sample of responses $U_1, \ldots, U_n$ is Fisher's

$$I_{U_1, \ldots, U_n}(\theta) = \frac{\partial}{\partial\theta}\left[E_\theta \ln\left[L(\theta;\, U_1, \ldots, U_n)\right]\right]^2$$

(e.g., Kendall & Stuart, 1979, Section 17.16). Because local independence is assumed in item response theory, the information in $U_1, \ldots, U_n$ is additive in the individual response variables. For the model in (1) it follows that

$$I(\theta) \equiv I_{U_1, \ldots, U_n}(\theta) = \sum_{i=1}^{n} I_i(\theta), \tag{2}$$

with

$$I_i(\theta) \equiv I_{U_i}(\theta) = P_i'(\theta) / \left\{ P_i(\theta)\left[1 - P_i(\theta)\right] \right\}^2. \tag{3}$$

For the Rasch model the latter simplifies into

$$I_i(\theta) = P_i(\theta)\left[1 - P_i(\theta)\right]. \tag{4}$$

One of the main reasons for using maximum likelihood scoring is the availability of $I(\theta)$ and $I_i(\theta)$ as (asymptotic) measures for the accuracy by which the test and the items measure the examinee's ability. It should be noted that these measures are dependent on the examinee's true ability value θ. For this reason they are known as the test and item information functions in IRT. A graphical display of the information functions for the items in Figure 2 is given in Figure 3. For more theory with respect to information functions the reader is referred to Birnbaum (1968, Chapter 17) or Lindgren (1976, Section 4.5.4.).

Information functions play a central role in

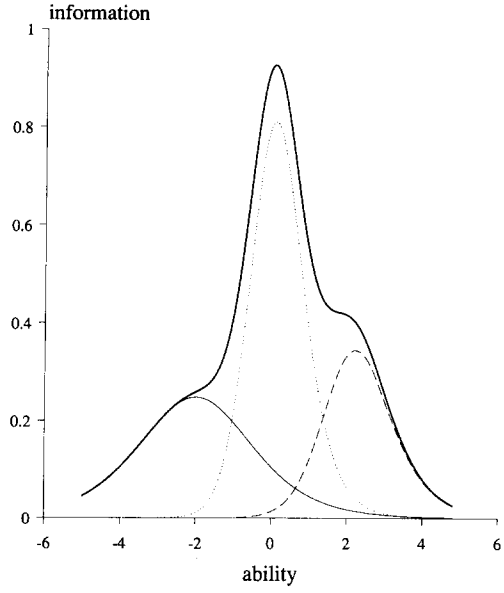

Figure 3. Information functions for the items in Figure 2 and their test information function

IRT-based test construction models. The basic idea is the following: (1) the test constructor specifies a target for the test information function values at some selected ability levels; and (2) test items are selected such that these target values are realized. The problem how to elicit a target for the test information function from a test constructor has been addressed in Kelderman (1986) and van der Linden and Boekkooi-Timminga (1989). In the former reference, the well-known asymptotic result on ML estimation is used to transform the information function into the standard error of estimation for the difference in ability between two examinees, and this metric is used to elicit a target information function. The procedure in the latter reference is explained below.

3. Models

Now some basic (mixed) 0–1 LP models for achievement test construction are considered, along with some of the constraints that may have to be included in these models to meet usual requirements in the practice of test administration.

3.1. Model of minimum test length

The first 0–1 LP model for test construction was proposed by Theunissen (1985). The goal of his model was to minimize the number of items in the test. Let $T(\theta_k)$ be the target for the value of the test information function at ability level θ_k, $k = 1, \ldots, K$, as specified by the test constructor. The test constructor is free to choose the number and spacing of the ability levels to guarantee a required precision. Define the decision variables x_i as

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

Then, the model can be formulated as the following constrained minimisation problem:

$$\min \quad \sum_{i=1}^{N} x_i \tag{5}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} I_i(\theta_k) x_i \geq T(\theta_k), \quad k = 1, \ldots, K, \tag{6}$$

$$x_i \in \{0, 1\}, \quad i = 1, \ldots, N. \tag{7}$$

Since (2) and (3) are well-behaved continuous functions the test information function generally will be above the target value in the range of interest, if the points θ_k are chosen close enough to one another. In practice, usually only three or four points are needed.

3.2. Maximin model

In the above model it is assumed that the test constructor does not want to have control of the test length and is able to specify a target for the test information in the required metric. A maximin model proposed by van der Linden and Boekkooi-Timminga (1989) circumvents these problems; because the test constructor now has to specify only the *relative* shape of the target by specifying values for parameters r_k, $k = 1, \ldots, K$. This can be done, for instance, by presenting the ability scale θ to the test constructor as a line, and to request him/her to distribute a number of chips (say, 100) over the values θ_k, $k = 1, \ldots, K$, such that they reflect the relative distribution of information wanted from the test. The value of r_k is then equal to the number of chips at θ_k. Now the maximin model can be given as

$$\max \quad z, \tag{8}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} I_i(\theta_k) x_i - r_k z \geq 0, \quad k = 1, \ldots, K, \tag{9}$$

$$\sum_{i=1}^{N} x_i = n, \tag{10}$$

$$x_i \in \{0, 1\}, \quad i = 1, \ldots, N, \tag{11}$$

$$z \geq 0. \tag{12}$$

In this model the test length is fixed by the test constructor. Because n is fixed, the model maximizes z such that the test information is just larger than $(r_1 z, \ldots, r_{Kz})$ uniformly in k.

In addition to the above two models, other models with target information functions have been proposed by Gademann (1987) and van der Linden and Boekkooi-Timminga (1989). Models based on classical test theory can be found in Adema and van der Linden (1989).

3.3. Practical constraints

In practice, various constraints with respect to the properties of the test may apply. Four examples of such constraints of interest are given as linear (in)equalities in the decision variables.

Although all items in the bank are assumed to measure the same skill or domain of knowledge, test items differ with respect to format or content aspects (e.g. sections in a text-book). In practice, constraints considering such aspects are often needed because test constructors want to control the selection of items in this sense. Actually, when unidimensionality holds, such constraints are of no importance to the measurement accuracy of the test, so their only purpose is to satisfy possible desires of the test constructors. Let V_j, $j = 1, \ldots, J$, be subsets of items in the bank from which the test constructor wants to select $n_j \leq n$ items, then the following constraints should be included in the model:

$$\sum_{i \in V_j} x_i = n_j, \quad j = 1, \ldots, J. \tag{13}$$

Of course, if the subsets form a partition of the item bank, (13) should be specified such that $\Sigma_{j=1}^{J} n_j = n$. The following example illustrates the use of (13): consider an item bank for testing French as a foreign language. This bank can be partitioned into subsets, for instance, with respect to content (e.g., vocabulary, grammar, or reading comprehension), format of the items (e.g., multiple choice, completion), or a behavioural taxonomy (e.g., knowledge of facts, applications of rules, or evaluation). It is assumed that all subsets measure the same language ability. Choosing values for n_j in (13), the composition of the test with respect to these dimensions is governed.

Another possible constraint has to do with the administration time needed for the test. Let t_i be the time needed in the population of examinees to solve item i. If only T minutes are available, the following constraint should be met:

$$\sum_{i=1}^{N} t_i x_i \leq T. \tag{14}$$

Some items may contain cues for the answers to other items in the bank. Such items should not be included in the same test. Let V_j, $j = 1, \ldots, J$, now indicate subsets of mutually exclusive items in the bank. The question whether such subsets will ever fit a response model is deliberately omitted. The problem is only raised to show that if such subsets should happen to fit the same model, then to prevent the test from containing more than one item of V_j, $j = 1, \ldots, J$, the following constraints could be included in the model:

$$\sum_{i \in V_j} x_i \leq 1, \quad j = 1, \ldots, J. \tag{15}$$

The opposite case can also occur, for instance, in the well-known format where the answers to some of the preceding items in the tests are needed to solve a later item. The presence of such items leads to the following linear constraints: Suppose V_j, $j = 1, \ldots, J$, indicates subsets of items in the bank for which the selection of one item implies the selection of all other items in the same set. Now it should hold

$$\sum_{i \in V_j} x_i = |V_j| x_{i_j}, \quad j = 1, \ldots, J, \tag{16}$$

where $|V_j|$ denotes the number of items in V_j and x_{i_j} is the decision variable of an arbitrary item in V_j. In addition to the above examples, various other constrains may apply in the practice of achievement testing. A complete review is given in van der Linden and Boekkooi-Timminga (1989).

Thus far, the case of constructing one test at a time was considered. However, in some applications more than one test has to be constructed, that should have a special relation to each other, for instance, be parallel. The optimal procedure in such applications is simultaneous test construction

(Boekkooi-Timminga, 1987), which can be attained via a straightforward modification of the model (see Example 2 below).

4. Computational procedure

In this section a heuristic based on some ideas proposed by Crowder, Johnson, and Padberg (1983) to speed up the branch-and-bound method is described. The heuristic is very useful for test construction applications. To aim at generality, 0–1 programming models of the following form are considered:

$$\text{maximize}\{\, c'x \mid Ax \leq b,\ x_j = 0 \text{ or } 1,\ j = 1, \ldots, n\}, \qquad (17)$$

where A is an $m \times n$ matrix, b is a vector of length m and c and x are vectors of length n.

The continuous optimal objective function will be denoted by Z_{LP} and the true lower bound on the 0–1 optimal objective function by Z_+.

If no feasible solution to (17) is known, the branch-and-bound method is initialized by assuming $Z_+ = -\infty$; but if the optimal value of the 0–1 objective function is known to be close to Z_{LP}, the method can, after solving the relaxation of (17), be initialized by $Z_+ = K_1 Z_{LP}$, where K_1 is a constant ($0 \ll K_1 < 1$).

Given the initialization it is clear that every 0–1 solution found during the search process has a value for the objective function between $K_1 Z_{LP}$ and Z_{LP}. So if K_1 is close to 1, the solution is good, and the branch-and-bound method can be stopped when the first feasible solution for (17) is found. In this way a good but not necessarily best solution is obtained. In most applications this is no problem because the coefficients in the model are estimates of certain unknown quantities and the difference between the exact solution and the one found can be made arbitrarily small. For instance, in the examples below K_1 was put equal to 0.995.

In the heuristic, the continuous optimal reduced costs, d_j, corresponding to variable x_j are used to fix nonbasic variables at the value 0 or 1:

(1) fix x_j to 0 if in the continuous solution $x_j = 0$ and $Z_{LP} - K_2 Z_{LP} < d_j$;

(2) fix x_j to 1 if in the continuous solution $x_j = 1$ and $Z_{LP} - K_2 Z_{LP} < -d_j$, where $K_2 < 1$. The above rules are applied after the continuous solution of the relaxation of (17) is found. The value of K_1 cannot be chosen as high as the value of K_2, because when specifying K_1 it should be certain that the value of the objective function for the solution of (17) be larger than $K_1 Z_{LP}$. If the value of K_1 or K_2 is too large, the decision tree is small, and it will not take much time before it is clear that no solution to (17) can be found. In such a case the values of K_1 and/or K_2 should be adjusted and the procedure be started anew.

The choice of K_1 and K_2 depends on the underlying item response model. For the Rasch model the information functions are more similar than for the three-parameter model. Therefore, K_1 and K_2 can be chosen closer to 1 for the Rasch model. Also, the practical constraints influence the choice of K_1 and K_2. In general, the less restrictive the 0–1 LP model, the closer to 1 K_1 and K_2 can be chosen. The choice of K_1 and K_2 will not always be appropriate, but this problem can be solved by adjusting the values of K_1 and/or K_2 as already stated.

5. Examples

Two examples are presented. The first example addresses the case of constructing a single test; a number of practical constraints are included in the model. The second example addresses the problem of constructing two parallel tests simultaneously.

The above heuristic was used to solve the models on a DEC-2060 computer. The modifications in the branch-and-bound strategy were introduced in the program LANDO (Center for Mathematics and Computer Science CWI, Amsterdam)

5.1. Example 1

An item bank for English was simulated. Six hundred items all fitting the Rasch model were considered (that is, the response model was the one in (1) with $a_i = 1$ and $c_i = 0$). The item difficulties, b_i, were drawn from the standard normal distribution. For the sake of illustration, the time in seconds required to administer each individual item was drawn from a uniform distribution with range [20,60]. The item bank was supposed to be

divided in three subsets of items each covering a different domain of content:

items 1–200: vocabulary items;

items 201–400: grammar items;

items 401–600: reading comprehension items.

The first 100 items of each subset were assumed to be of the multiple-choice type; the other items were essay items.

Now suppose a test constructor wants to have a test with the following specifications:

(1) At ability levels $\theta_1 = -1$, $\theta_2 = 0$, and $\theta_3 = 1$ the information in the test should be approximately equal.

(2) The test should contain 14 vocabulary, 16 grammar, and 10 reading comprehension items.

(3) Exactly 16 multiple-choice items and 24 essay items should be included in the test.

(4) The test administration time is not allowed to exceed 1500 seconds.

(5) No more than one item may be selected from the topic covered by the first ten items in the item bank.

The following model was used to realize the test:

$$\max \quad z, \tag{18}$$

$$\text{s.t.} \quad \sum_{i=1}^{600} I_i(\theta_k) x_i - z \geq 0, \quad k = 1, 2, 3, \tag{19}$$

$$\sum_{i=1}^{200} x_i = 14, \tag{20}$$

$$\sum_{i=201}^{400} x_i = 16, \tag{21}$$

$$\sum_{i=401}^{600} x_i = 10, \tag{22}$$

$$\sum_{i=1}^{100} x_i + \sum_{i=201}^{300} x_i + \sum_{i=401}^{500} x_i = 16, \tag{23}$$

$$\sum_{i=101}^{200} x_i + \sum_{i=301}^{400} x_i + \sum_{i=501}^{600} x_i = 24, \tag{24}$$

$$\sum_{i=1}^{600} t_i x_i \leq 1500, \tag{25}$$

$$\sum_{i=1}^{10} x_i \leq 1, \tag{26}$$

$$x_i \in \{0, 1\}, \quad i = 1, 2, \ldots, 600, \tag{27}$$

$$z \geq 0. \tag{28}$$

The results for these specifications are shown in Table 1. The values of z are the lower bounds to the test information function maximized in (18). As can be seen, the value for the 0–1 problem is only slightly smaller than the one for the relaxed problem. The times needed for reading the input file, for the initialization, and for writing to the output file were not included in the CPU-times in Table 1. The CPU-times for solving the relaxed and the 0–1 problems are given under the heads 'Relaxed' and '0–1', respectively. The good news is that, even for K_1 as close to 1 as 0.995, the proposed heuristic gives a solution in 26.5 seconds, whereas a 0–1 problem of this size generally cannot be solved in realistic time. Similar results were obtained in a large simulation study in which the objective function and constraints were varied (Adema, 1988).

5.2. Example 2

Tests are defined to be parallel if their information functions are identical (Samejima, 1977). Parallel tests are used, for instance, when secrecy problems prevents the tester from using the same test for different groups of examinees. Suppose two parallel tests have to be selected from an item bank of 300 items fitting the Rasch model. The item difficulty values b_i were drawn from a standard normal distribution.

A 0–1 LP model for the simultaneous construction of parallel tests was formulated. Let the decision variables x_{it} indicate whether or not item i is selected for test t. In the model below the objec-

Table 1
Results of Example 1

z		Test information			CPU-time (sec)	
Relaxed	0–1	$I(-1)$	$I(0)$	$I(1)$	Relaxed	0–1
7.861	7.859	7.859	9.982	7.859	16.7	9.8

$K_1 = 0.995$; $K_2 = 0.9999$.

tive function z in (29) maximizes the total amount of information in the two tests, subject to the constraints in (30) that for each test the information function reflects the shape as specified by r_k. By maximizing this common lower bound z, the obtained lower bounds for the test information function values of both tests constructed will be close. In the example, r_k was set equal to 1 for $\theta_1 = -1$, $\theta_2 = 0$ and $\theta_3 = 1$. Also, each test should contain 30 items, and the tests should not overlap.

The model was as follows:

$$\max \quad z, \tag{29}$$

$$\text{s.t.} \quad \sum_{i=1}^{300} I_i(\theta_k) x_{it} - r_k z \geq 0,$$

$$k = 1, 2, 3, \quad t = 1, 2, \tag{30}$$

$$\sum_{i=1}^{300} x_{it} = 30, \quad t = 1, 2, \tag{31}$$

$$x_{i1} + x_{i2} \leq 1, \quad i = 1, \ldots, 300, \tag{32}$$

$$x_{it} \in \{0, 1\}, \quad i = 1, \ldots, 300, \quad t = 1, 2, \tag{33}$$

$$z \geq 0. \tag{34}$$

In Table 2, the values of z for the relaxed as well as the 0–1 problem are given, together with the values of the test information functions and the CPU-times. As noted earlier, tests are considered to be parallel if they have identical information functions. Comparing the values for the information functions in Table 2, it can be concluded that the two tests were parallel indeed. Again, it can be seen that for large values of K_1 as 0.995, the heuristic produces excellent results in practical CPU-times.

6. Conclusion

In the construction of tests from item banks mathematical programming plays an important role. In this paper some of the most promising test construction models were highlighted. A heuristic was developed to solve large-scale applications. As the examples show, the heuristic performs well. Similar results were obtained in a larger study (Adema, 1988).

It can be concluded that the application of 0–1 linear programming to test construction problems allows test agencies to computerize their services. At present prototypes of test construction systems are developed to try out their usage in the practice of achievement testing.

Acknowledgement

The Netherlands Organization for Scientific Research (N.W.O.) is gratefully acknowledged for funding part of this project. Part of this research was conducted while Ellen Boekkooi-Timminga was supported by a PSYCHON-grant of this organization (560-267-001), awarded to Dr. W.J. van der Linden.

References

Adema, J.J. (1988), "A note on solving large-scale zero–one programming problems", Research Report 88-4, Department of Education, University of Twente, Enschede.

Adema, J.J., and van der Linden, W.J. (1989), "Algorithms for computerized test construction using classical parameters", *Journal of Educational Statistics* 14, 279–290.

Birnbaum, A. (1968), "Some latent trait models and their use in inferring an examinee's ability", in: F.M. Lord and M.R. Novick (Eds.), *Statistical Theories of Mental Test Scores*, Addison-Wesley, Reading, MA.

Table 2
Results of Example 2

Test	z		Test information			CPU-time (sec)	
	Relaxed	0–1	$I(-1)$	$I(0)$	$I(1)$	Relaxed	0–1
1	5.882	5.862	5.898	7.436	5.862	37.3	28.1
2			5.876	7.424	5.877		

$K_1 = 0.995$; $K_2 = 0.9999$

Boekkooi-Timminga, E. (1987) "Simultaneous test construction by zero–one programming", *Methodika* 1, 101–112.

Boekkooi-Timminga, E. (1990), "Parallel test construction from IRT-based item banks", *Journal of Educational Statistics* 15, 129–145.

Crowder, H., Johnson, E.L., and Padberg, M. (1983), "Solving large scale zero-one programming problems", *Operations Research* 31, 803–834.

Fischer, G.H. (1974), *Einführung in die Theorie Psychologischer Tests: Grundlagen und Anwendungen*, Verlag Hans Huber, Bern.

Gademann, A.J.R.M. (1987), "Item selection using multiobjective programming", OIS-project Report no. 01, National Institute for Educational Measurement (Cito), Arnhem.

Hambleton, R.K., and Swaminathan, H. (1985), *Item Response Theory: Principles and applications*, Kluwer–Nijhoff, Boston.

Kelderman, H. (1986), "A procedure to assess information functions for the construction of tests measuring multiple traits", in: G.R. Buning, T.J.H.M. Eggen, H. Kelderman, and W.J. van der Linden (eds.), *Het Gebruik van het Raschmodel voor een Decentraal Toetsservicesysteem* [The Use of the Rasch Model for a Decentralized Testing Service System] (Rapport 86-3), University of Twente, Department of Education, Enschede.

Kendall, M., and Stuart, A. (1979), *The Advanced Theory of Statistics*, Vol. 2, Griffin, London.

Lindgren, B.W. (1976), *Statistical Theory*, 3rd ed., Macmillan, New York.

Lord, F.M. (1980), *Applications of Item Response Theory to Practical Testing Problems*, Lawrence Erlbaum, Hillsdale, N.J.

Rasch, G. (1960), *Probabilistic Models for some Intelligence and Attainment Tests*, Nielsen and Lydiche, Copenhagen.

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. (1988), "Optimizing incomplete sample designs for item response model parameters", Research Report 88-5, Department of Education, University of Twente, Enschede.

van der Linden, W.J., and Boekkooi-Timminga, E. (1989), " A maximin model for test design with practical constraints", *Psychometrika* 54, 237–247.

van Thiel, C.C. and Zwarts, M.A. (1986), "Development of a testing service system", *Applied Psychological Measurement* 10, 391–403.

Yen, W.M. (1983), "Use of the three-parameter model in the development of a standardized achievement test", in: R.K. Hambleton (ed.) *Applications of Item Response Theory*. Educational Research Institute of British Colombia, Vancouver B.C.