Dealing with multiple criteria in test assembly

Bernard P Veldkamp

Research Center for Examination and Certification

University of Twente - The Netherlands

Mariagiulia Matteucci

Statistics Department

University of Bologna - Italy

Requests for information can be send to: Bernard Veldkamp, University of Twente, PO box 217, 7500 AE Enschede, The Netherlands. E-mail: b.p.veldkamp@utwente.nl

Abstract

It is quite common that tests or exams are being used for more then one purpose. First of all, they are used to measure the ability of the students in a reliable manner. Besides, they can be used for pass/fail decisions or to predict future behavior of the candidate, like future job behavior or academic performance. The question remains how to assemble a test that can be used for all these different purposes, that is, how to assemble a multi-objective test. Besides, multiple objectives can result from different purposes, but also from the way test specifications have been implemented. For the WDM-model, for multidimensional IRT, for Cognitive Diagnostic CAT, but also for infeasibility analysis, multiple objective test assembly problems have to be solved.

In this paper, a 2-stage method is presented for dealing with multiple objectives in test assembly. In the normalization stage, all objectives are brought on a common scale. In the valorization stage, the different objectives are being compared and related to each other. The method is applied to a Guidance Test developed at the University of Bologna, and a comparison is made with more traditional single objective test assembly methods. The results clearly demonstrate the importance and relevance of multi-objective test assembly.

Keywords: automated test assembly, multiple criteria, multiple objectives, simulated annealing, guidance test.

Introduction

More and more often, it happens that a single test is used for several purposes. For example, in the Netherlands, the final examination at the end of High school is used to make pass/fail decisions for individual examinees, to evaluate the performance of the school, and to predict future performance of the examinee. Using a test for multiple purposes might seem rather efficient, but the question remains whether it is a valid approach to apply grades that have been obtained for making, for example, pass/fail decisions in an entirely different setting like evaluating school performance. From a psychometric point of view, several objections can be formulated, but psychometrics might also provide the answers.

In test assembly, items are assigned to a test from an item pool. The pool contains many items with different characteristics or attributes, for example, content specification, gender orientation, average response time, or word counts, but also psychometric features like item information, item difficulty, or discriminating power. In the early days of testing, items were assigned to tests by hand. They were picked from the item pool until an initial test was assembled. Then, the lengthy and boring process of improving the composition of the test by interchanging items in the test with items in the pool began, until the test assembler was convinced that the test performed well enough with respect to its specifications. Nowadays, items are often assigned by computer programs that are based on mathematical programming techniques (Armstrong, Jones, and Wang, 1995, Belov, & Armstrong, 2005, Luecht, & Hirsch, 1992, Stocking, & Swanson, 1993, van der Linden, 2005, van der Linden, & Boekkooi-Timminga, 1989). In large testing programs, the pools typically consist of several thousands of items, and tests are specified by hundreds of constraints, so it is hard to find a test that matches all the specifications and practically impossible to find the test that matches them best.

Characteristic for mathematical programming is the formulation of a single objective function that is maximized over a set of admissible tests, where the test specifications determine whether a test is admissible or not. In practice, a wide variety of objective functions is used, and many different kinds of test specifications are imposed on tests. However, for several applications, the format of a single objective function turns out to be too restrictive.

When a test is designed to be used for several purposes, several objective functions have to be taken into account. Besides, several objectives might also be necessary for technical reasons. For example, for applications of the Weighted Deviation Model (Stocking & Swanson, 1993), for Multidimensional IRT (Veldkamp & van der Linden, 2002), for Cognitive Diagnostic CAT (Cheng & Chang, 2007) or for infeasibility analysis (Huitzing, 2004, Huitzing, Veldkamp, & Verschoor, 2005) several objectives have to be combined.

The aim of this study is to develop and compare methods for combining the different, sometimes conflicting, goals of testing. The methods are applied to the Guidance Test where the following objectives interact. The test has to be short, measurement precision has to be optimized for different content areas, and classification decisions about which faculties suit the students best, have to be made. Finally, conclusions about how to deal with multiple objectives in automated test assembly are formulated.

Dealing with test specifications

The input of any model for test assembly is a set of specifications that describes the features of the test. A general taxonomy of test specifications can be found in van der Linden (2005), where specifications are classified both according to the *nature* and to the *level* of the constraint. In this taxonomy, fifteen different categories are distinguished. The 2-dimensional classification table has been defined as {categorical, quantitative, logical}×{item level, item set level, testlet level, test level, multiple test level}. Test specifications can be modeled, either as constraints or as objectives (van der Linden, 2005).

When a specification is modeled as a constraint, it implies that this specification must be met during the test assembly process (Timminga, 1998). The specification is said to be 'mandatory'. Constraints can be defined as

$$\sum_{i \in G} V_i x_i \leq t, \qquad (1)$$

where x_i denotes whether the item is selected ($x_i = 1$) or not ($x_i = 0$), the summation ranges over those items contained in the object group G, t is some target value to be met. V_i is determined by the type of the specification imposed. For example, for test composition specifications, V is the identity vector, and the constraint just counts how many items in G are selected for the test. For statistical specifications, V denotes the contribution of each item. Target values can be formulated for different applications. It could either be a lower boundary, the target of the specifications, or an upper boundary (see Table 1).

Insert Table 1 at about here

Test specifications can also be modeled as objectives. These objectives can be used for two different purposes. The first purpose is for specifications related to the goals of testing. When a goal is to maximize test information or to minimize test length, these specifications can be formulated as

$$\max \sum_{i \in G} V_i x_i, \text{ or } \quad \min \sum_{i \in G} V_i x_i, \tag{2}$$

where V relates to the attribute being optimized.

The second purpose is to handle specifications as desired properties. The optimization process results in a test with the specification as close to the pre-defined target as possible. Goal programming techniques (Veldkamp, 1999) can be applied to account for this purpose. When specifications are seen as desired properties, the objective for specification j can be defined as

$$\min d_j \tag{3}$$

s.t.

$$\left| \sum_{i \in G} V_i x_i - t_j \right| \leq d_j . \tag{4}$$

where d_j is the deviation from the desired target t_j. It depends on preference of the user what kind of distance metric should be favored. Manhattan distance (p=1), Euclidean distance (p=2) or Chebyshev distance ($p \rightarrow \infty$) are generally applied. Manhattan distance minimizes the sum of the deviations, and is applied in most test assembly models (Stocking, & Swanson, 1993). Euclidian distance results in a quadratic optimization problem, with is rather complicated to solve when it is combined with a set of mandatory constraints. Chebyshev distance minimizes the maximum deviation. The Chebyshev distance emphasizes justice and balance rather than straightforward optimization, by focusing on the specification with largest deviation. This metric is successfully applied in van der Linden & Boekkooi-Timminga (1989), for minimizing the maximum deviation from a target information function. For the majority of test assembly problems, application of the Chebyshev distance seems less favorable. These models typically consist of large numbers of specifications, and focusing on the largest deviation implies that deviations of the remaining objectives are bounded from above, instead of optimized.

How to deal with different objectives?

It is important to recognize that deviations measured in different units cannot be summed directly due to the phenomenon of incommensurability. Weight factors have to be added to bring the objectives on a common scale. Besides, weight factors can also be used because users might assign different priorities to different specifications. In other words, weight factors are composed of two components destined to represent two different roles.

The first one is "normalization" meaning to bring all deviations to a common scale based on the degree of proximity to the goal. The second is "valorization" reflecting the decision-maker's priority structure. The normalization should be carried out first. Once all the objectives are on a common scale, different priorities can be assigned to the normalized objectives.

Normalization

In order to bring different specifications on a common scale, each unwanted deviation can be multiplied by a normalization constant to allow direct comparison. Many normalization procedures have been proposed in the literature (Romero, 1991). Popular choices for normalization constants are the target value of the corresponding objective or the range of the corresponding objective.

Target value. One way to bring all the objectives unto a common scale is to divide the deviation by its target. In fact, this transformation turns all deviations into percentages. The transformation formula is given by

$$d_j' = \frac{d_j}{t_j}. \tag{5}$$

Range of the objective. The other way is to normalize deviations based on their range. In this case, all deviations are transformed into the same interval [L_{min} , L_{max}] based on the difference between their maximum deviation $d_{j,max}$ and minimum deviation $d_{j,min}$. The transformation formula is given by

$$d_j' = \frac{\left(d_j - d_{j,\min}\right)(L_{\max} - L_{\min})}{(d_{j,\max} - d_{j,\min})} + L_{\min}. \tag{6}$$

This transformation looks pretty complicated, but when all objectives are transformed to the interval [0,1], and it is taken into account that the minimum deviation is zero by definition, the formula boils down to

$$d_j' = \frac{\left(d_j\right)}{\left(d_{j,\max}\right)}. \tag{7}$$

About this transformation, some remarks can to be made. Since $d_{j,max}$ is the worst possible performance with respect to target *j* of any test that meets the mandatory specifications, application of this transformation implies that the worst possible test sets the standards. Although this results from a straightforward application of this rule, it does not sound very appealing, and it will probably not convince many future applicants. An alternative would be to replace the maximum deviation by the deviation observed in a reference test.

Valorization

Once all objectives have been transformed to a common scale, different weights can be assigned to reflect the priorities of the user. For every test specification a lower bound, a target value, and an upper bound can be defined. But not all of them have to be equally important. It might be very important that examinees finish a test within three hours (upper bound), while it would be just nice when they finish it in about two hours (target value). Besides, priority can be different for different specifications. Constraints related to average p-values, might be more important than those about word count, or answer keys.

To rate objectives as very important, just nice, important, or utterly unimportant, etc., is a complicated task. The Analytic Hierarchy Process (Saaty, 1990) has been proposed to deal with this kind of problems. In the AHP, a matrix of pair wise comparisons is composed. The eigenvector of this comparison matrix can be applied successfully for ranking the priorities. Imagine a specification dealing with the number of algebra items in a test. For example, let the number of algebra items be at least two (t_1=2), at most 7 (t_3=7), and the preferred number of algebra items in a test is six (t_2=6). The results of the pair wise comparison might be that the target value is twice as important as the lower bound, and four times as important as the upper bound. The lower bound is three times as important as the upper bound. For the comparison matrix (see Table 2), the eigenvector can be calculated to be (0,3196; 0,5584; 0,1220). These eigen values can be applied as valorization weights in the goal programming model.

Insert Table 2 at about here

A different method is described in van der Linden & Boekkooi-Timminga (1989), where a finite set of marbles has to be distributed over the different objectives to rate their importance. Imagine that ten marbles are available for prioritizing the objectives related to the specification about the number of algebra items in a test. After distributing the marbles over the different objectives, a possible outcome is shown in Figure 2.

Insert Figure 2 at about here

When models consist of hundreds of constraints, both of these valorization methods become intractable, and less demanding ranking or rating methods are needed. In practical settings it might be usefull just to assign a high, a medium, or a low priority to every constraint. These priorities might be translated into double, normal of one-half valorization weights for every objective.

Different valorization methods can be applied at different levels. At test specification level, the high/medium/low method seems to be the only reasonable choice due to the large number of specifications in most test assembly models. Within each specification, either one of the AHP, Marbles or high/medium/low methods might be applied to assign valorization weights v_{jk}.

Multilevel structure

One of the important features of test assembly models is that for any test specification, three different targets can be set; t_{j1} for the lower bound, t_{j2} for the target value, and t_{j3} for the upper bound. These three objectives are nested within the test specification, because they already are on the same scale. In this way, a multilevel structure of test specifications can be defined. At the first level, the lower bound, target value, and upper bound are defined. At the second level, different test specifications are distinguished.

When test specifications deal with, for example, test information, even a third level might be added. Test information is a function of the ability level of the examinee. Typically, the test information function is supposed to be within a certain bandwidth of the target information function (See Figure 1). When this constraint is implemented in a test assembly model (van der Linden & Boekkooi-Timminga, 1989), targets have to be set for a finite number of abilities. So, for a number of theta values, a lower bound, a target value, and an upper bound might be imposed.

Insert Figure 1 at about here

As a consequence, normalization and valorization procedures have to be applied in a multilevel context with nested sets of constraints.

Modifications of normalization procedure. Both the target value procedure and the procedure don't take the nested nature of test assembly objectives into account. For every test specification, the nested objectives deal with the same object group G and are on the same scale. Because of this, they need a common scaling factor.

When the target value procedure is applied, a rather straightforward way would be to average the values of the related targets, when more than one target is set related for a specification. The resulting transformation function can be defined as

$$d_{jk}' = \frac{d_{jk} \sum_{k=1}^{3} I_{jk}}{\sum_{k=1}^{3} I_{jk} t_{jk}} \tag{8}$$

where I_{jk} indicates whether a target is set for the lower bound (k=1), the target value (k=2) or the upper bound (k=3) of specification j.

For the range normalization procedure, the specific nature of test assembly objectives provides the possibilities for a different implementation, since the targets for the upper and lower bound can be applied to define a common scaling factor. A natural range for the deviations is defined by the interval between the upper and lower bound of the objective. The resulting transformation function for the range procedure can be defined as

$$d_{jk}' = \frac{\left(d_{jk}\right)}{\left(t_{j3} - t_{j1}\right)}. \tag{9}$$

This procedure assumes the definition of targets for both the lower and the upper bound of an objective. Whenever a test specialist only defines one of them, the range could be defined by two times the difference between either the lower or the upper bound and the target value of the objective.

When the normalization and validation procedures are combined, for every deviation d$_{jk}$ belonging to specification j a weight factor w_{jk} can be formulated, either via a combination of the target value normalization and a valorization procedure or via the range normalization procedure and a valorization procedure. The resulting weight factors can therefore be formulated as

$$w_{jk} \in \left\{ \frac{v_{jk}\sum_{k=1}^{3} I_{jk}}{\sum_{k=1}^{3} I_{ji} t_{jk}} \;,\; \frac{v_{jk}}{\max_{k}\{d_{j,ref,k}\}} \right\} \qquad \text{for all } (j,k). \qquad (10)$$

Numerical Example

Data in this study come from a Guidance Test developed at the University of Bologna (Matteucci, 2007). This test was developed to help students in their choice of a adequate faculty at the University of Bologna, and to prevent them from dropping out early due to lack of competence. A few years ago, students of the 4th and 5th year of high secondary school (about 17-19 years old) could only fill in a psychological test to verify their competences to enroll certain faculties, but they could not get deep information about their current knowledge. Now they can visit the Guidance Service web site (http://orientaonline.unibo.it/) fill out a faculty-specific online test that consists of a general part, measuring general culture, and a faculty specific part.

Items. The guidance test consists of 30 items in the general part and 20 items for each faculty specific part. For the purpose of this study, the 2-PLM (Lord,1980) was fitted to the items in the general part, and to the items in each faculty specific part separately. Besides, for the items of the general part, the variance σ_i^2, the item-test correlation ρ_{iX}, and the item-faculty correlations ρ_{iJ} are known for all items *i* and scores on the faculty specific parts *J*. For the items in the general part, also a content classification is provided (current news, civic culture, general humanistic, geography, and technical/scientific). For all items, the answer keys are known.

Test assembly problem. A short form of a Guidance test developed within the University of Bologna has to be assembled. The length of the general part has to be reduced to only 10 items. In this problem, two objectives have to be taken into account. First of all, measurement precision of the general part has to be maximized. The second objective is to maximize predictive validity, where the score on the faculty specific part is the criterion to be predicted. Moreover, for every content classification the number of items in the test is fixed.

Design of the study. First, a short form with maximum information is assembled. After that, short forms are assembled that maximize predictive validity for each faculty. In this study, we focus on the Language, Politics and Arts faculties. Finally, both objectives are combined and for every faculty a short form of the general part is assembled that optimizes both measurement precision and predictive validity. Since the predictive validity of a test is non-linear in the items, Simulated Annealing (Veldkamp, 1999, van der Linden, Veldkamp, & Carlson, 2004) was applied for test assembly. The presence of content constraints forced us to modify the standard implementation of the heuristic to prevent problems due to infeasibility (see e.g. Huitzing, Veldkamp, &

Verschoor, 2005) of the resulting tests. In the standard implementation of Simulated Annealing the following iterative procedure is applied. A group of items is selected from the pool that meets the constraints. For this test the value of the objective function, e.g. the predictive validity or the information, is calculated. Then a random swap from one item in the incumbent test with one item in the pool is applied. The performance of the new test is calculated. If this performance is better, the new test becomes the incumbent test. If the performance is worse, the new test becomes the incumbent test with a certain probability. This probability decreases during the process, and the heuristic terminates when the probability that a worse test is accepted has decreased below a pre-specified number. In our modified version of the heuristic, we did not perform a random swap of items. Instead, we restricted the items to be swapped to those having the same content classification. In this way, infeasibility problems were prevented.

First, a short form will be assembled that maximizes Fisher Information. Then, a short form will be assembled that maximizes predictive validity. Finally, a short form will be assembled that combines both objectives.

Results

Model 1: short form with maximum information

Following van der Linden (2005; p. 114), the following test assembly problem has to be solved:

$$\max y \tag{11}$$

subject to

$$\sum_{i=1}^{30} I_i(\theta_k)x_i \geq y \quad \theta_k \in \{-1,0,1\} \tag{12}$$

$$\sum_{i=1}^{30} x_i = 10 \tag{13}$$

$$\sum_{i \in V_c} x_i \geq n_c \quad \forall c \tag{14}$$

$$x_i \in \{0,1\} \tag{15}$$

Where (11) and (12) maximize the information in the test, (13) defines the test length, (14) accounts for an equal distribution of items over the content classes *c*, and (15) is a technical constraint that defines that either an item is selected ($x_i = 1$) or not selected ($x_i = 0$).

Items 2, 3, 4, 5, 7, 10, 12, 20, 21, and 29 were selected, and the resulting values for the test information function (I_{test} = 1,94) and the predictive validities ($\rho_{t,language}$= 0,05; $\rho_{t,politics}$=0,23 ; $\rho_{t,arts}$=0,09) are shown in Figure 3.

Model 2: short form with maximum predictive validity

Following van der Linden (2005; p. 118), the following test assembly problem has to be solved for each faculty *J*:

$$\max \frac{\sum_{i=1}^{30} \sigma_i \rho_{iY} x_i}{\sum_{i=1}^{30} \sigma_i \rho_{iX} x_i} \tag{16}$$

subject to

$$\sum_{i=1}^{30} x_i = 10 \tag{17}$$

$$\sum_{i \in V_c} x_i \geq n_c \quad \forall c \tag{18}$$

$$x_i \in \{0,1\}, \tag{19}$$

where (16) and maximizes the predictive validity of the test. For Language, items 2, 4, 8, 10, 12, 15, 19, 21, 25, and 29 were selected. For Politics, items 1, 3, 4, 9, 10, 18, 21, 23, 26, and 27 were selected. For Arts, items 5, 7, 11, 12, 13, 19, 21, 25, 28, and 29 were selected. The resulting values for the test information functions ($I_{language}$ = 1,52; $I_{politics}$ = 1,40; I_{arts} = 1,23) and the predictive validity ($\rho_{t,language}$= 0,13; $\rho_{t,politics}$=0,27 ; $\rho_{t,arts}$=0,18) for the different faculties are also shown in Figure 3.

Model 3: combined objectives

The range approach, Equation 6, was applied to normalize both objectives. The test information function *I* falls in the interval [0,I_{pool}], whereas the predictive validity was falls in the interval [-1,1]. Both were transformed to the interval [0,1]. To valorize both objectives, content experts valued the test information twice as important as the predictive validity. Following Veldkamp (1999), the following test assembly problem has to be solved for each faculty *J*:

$$\max \frac{y}{I_{pool}} + \frac{w_{pv}}{2} \cdot \frac{\sum_{i=1}^{30} \sigma_i \rho_{iY} x_i}{\sum_{i=1}^{30} \sigma_i \rho_{iX} x_i} \tag{20}$$

subject to

$$\sum_{i=1}^{30} I_i(\theta_k) x_i \geq y \quad \theta_k \in \{-1,0,1\} \tag{21}$$

$$\sum_{i=1}^{30} x_i = 10 \tag{22}$$

$$\sum_{i \in V_c} x_i \geq n_c \quad \forall c \tag{23}$$

$$x_i \in \{0,1\}, \tag{24}$$

where the weighting factor w_{pv} accounts for the valorization. For Language, items 2, 4, 5, 8, 10, 12, 19, 21, 25, and 29 were selected. For Politics, items 4, 5, 7, 10, 11, 12, 13, 19, 21, and 29 were selected. For Arts, items 1, 3, 4, 9, 10, 21, 23, 26, 27, and 29 were selected. The resulting values for the test information function ($I_{language}$ = 1,82; $I_{politics}$ = 1,72; I_{arts} = 1,54) and the predictive validities for the different faculties ($\rho_{t,language}$= 0,17; $\rho_{t,politics}$=0,27 ; $\rho_{t,arts}$=0,18) are also shown in Figure 3.

Insert Figure 3 at about here

In Figure 3 it can be seen how the different models perform with respect to the different objectives. The test resulting from Model 1 performs good with respect to the objective of high measurement precision, but the resulting predictive validity of the test w.r.t. the different faculty specific tests is pretty low. In Model 2, the focus is entirely on maximizing predicitive validity. For all faculties, the predictive validity increased considerably. Finally, in Model 3 both objectives were taken into account. For all three faculties, the amount of information increased at the costs of a small decrease in predictive validity.

Discussion

Tests are often used for several purposes. In this paper, a method was developed that facilitates handling of multiple objectives in test assembly. Besides, the method can also be applied to deal with multiple objectives due to the way test specifications are being handled. The clear distinction between normalization and valorization emphasizes the control that test committees or test assemblers have over the test assembly process.

In the numerical example, application of the method is illustrated (Model 3). Besides, the impact of differences in objectives is shown. The overlap between the tests resulting from models that maximizes information (Model 1), and models that maximize predictive validity (Model 2) is 50% or less, even in this case, where the item pool consists of only 30 items. The tests that maximize predictive validity for the different faculties (Model 2) overlap even less. The numerical example therefore not only illustrates the use of the methods, but also demonstrates the need. Only a well elaborated valorization process enables a test assembler or the testing committee to account for the results of the test assembly process. The technology is there, the next step is to implement it.

Besides, this paper also illustrates why one should be careful with the use of results of existing tests for new purposes. The results come from tests that have been developed for a given purpose, under a number of test specifications. The validity of the results is related to this purpose. In the empirical example, it can be seen what happens when the results of a test that was assembled to maximize information, is used to predict future performance. For all three faculties, the predictive validity of the test resulting from model 1 is very low. Even thought the short forms resulting from model 1 provide maximum information about the students, i.e. they reliably measure general knowledge, these tests are hardly useful for predicting future results. So when results from final examinations in the Netherlands are used for entrance decisions of universities, one has to be sure these examination results have enough predictive power, otherwise a selection test might be a more valid instrument for these decisions.

Finally, from a methodological point of view, several choices were made in this paper. The simulated annealing heuristic was used to assembly tests, even though in van der Linden (2005, p. 119) an iterative approach is described to handle predictive validity in a 0-1 linear programming context. The simulated annealing heuristic is a very general method for dealing with non-linear objectives, but in its general form it might result in tests that do not meet the specifications. On the other hand, Veldkamp (2002) demonstrated that transforming a non-linear problem in a linear problem might not always be very successful. Other heuristical methods could be applied in case of non-linearity. Recently, Genetic Algorithms (Verschoor, 2007, Finkelman, Kim, & Roussos, 2009) have been applied successfully in various test assembly problems.

Acknowledgements

We would like to thank the Guidance Service of the University of Bologna for allowing us to use their data for this study.

References

Armstrong, R.D., Jones, D.H., & Wang, Z. (1995). Network optimization in constrained standardized test construction. *Applications of Management Science, 8,* 189-212.

Bagozzi, R.P., Davis, F.D., & Warshaw, P.R. (1992). Development and test of a theory of technological learning and usage. *Human Relations, 45(7),* 660-686.

Belov, D., & Armstrong, R.D. (2005). Monte Carlo test assembly for item pool analysis and extension. *Applied Psychological Measurement, 29,* 239-261.

Bennett, P. D. (1995, 2nd edition). *AMA dictionary of marketing terms*. Chicago: American Marketing Association.

Cheng, Y., & Chang, H-H. (2007). *The maximum dual information method for cognitive diagnostic CAT.* Paper presented at the GMAC 2007 Computerized Adaptive Testing Conference, Minnesota (MN), June 7-8, 2007.

Davey, T., Steffen, M., Jodoin, M.J., & O'Hare, D. (2007). *Implementing a fully general model for automated test assembly.* Princeton, NJ: ETS.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13(3),* 319-340.

Finkelman, M., Kim, W., & Roussos, L.A. (2009). Automated test assembly for cognitive diagnostic assessment using a genetic algorithm. *Journal of Educational Measurement, 46,* 273-292.

Huitzing, H.A. (2004). Solving infeasible linear programming test assembly models. *Journal of Educational Measurement, 41,* 175-192.

Huitzing, H.A., Veldkamp, B.P., & Verschoor, A.J. (2005). Infeasibility in automated test assembly models: a comparison study of different methods. *Journal of Educational Measurement, 42,* 223-243.

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

Luecht, R.M., & Hirsch, T.M. (1992). Item selection using an average growth approximation of target information functions. *Applied Psychological Measurement, 16,* 41-51.

Matteucci, M. (2007). *Item response theory models for the competence evaluation: towards a multidimensional approach in the University guidance*. Unpublished PhD thesis, Statistics Department, University of Bologna (Italy).

Romero, C. (1991). *Handbook of critical issues in goal programming*. Oxford: Pergamon Press.

Saaty, T.L. (1990). How to make a decision: the analytic decision process, *European Journal of Operational Research, 48,* 9-26.

Stocking, M.L., & Swanson, L. (1993). A method for severely constrained item selection in adaptive testing. *Applied Psychological Measurement, 17,* 277-292,

Timminga, E. (1998). Solving infeasibility problems in computerized test assembly. *Applied Psychological Measurement, 22*, 280-291.

van der Linden, W.J. (2005). *Linear models for optimal test design*. New York: Springer.

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

van der Linden, W.J., Veldkamp, B.P., & Carlson, J.E. (2004). Optimizing balanced incomplete block designs for educational assessments. *Applied Psychological Measurement, 28,* 317-331.

Veldkamp, B.P. (1999). Multiple objective test assembly problems. *Journal of Educational Measurement, 36*, 253-266.

Veldkamp, B.P. (2002). Multidimensional constrained test assembly. *Applied Psychological Measurement, 26,* 133-146.

Veldkamp, B.P., & van der Linden, W.J. (2002). Multidimensional constrained adaptive testing. *Psychometrika, 67*, 575-588.

Verschoor, A.J. (2007). *Genetic Algorithms for Automated Test Assembly.* Unpublished Doctoral Thesis, University of Twente, Enschede, The Netherlands.

Table 1

Overview of constraints and objectives belonging to specification *j*

	Constraint	Objective
lower bound	$\sum_{i\in G} V_i x_i \geq l_j$	$\max\{l_j - \sum_{i\in G} V_i x_i, 0\} \leq d_{j1}$
target value	$\sum_{i\in G} V_i x_i = t_j$	$\left\lvert \sum_{i\in G} V_i x_i - t_j \right\rvert \leq d_{j2}$
upper bound	$\sum_{i\in G} V_i x_i \leq u_j$	$\max\{\sum_{i\in G} V_i x_i - u_j, 0\} \leq d_{j3}$

Table 2

Pair wise comparison matrix

	Lower bound	Target value	Upper bound
Lower bound	1/1	1/2	3/1
Target value	2/1	1/1	4/1
Upper bound	1/3	1/4	1/1

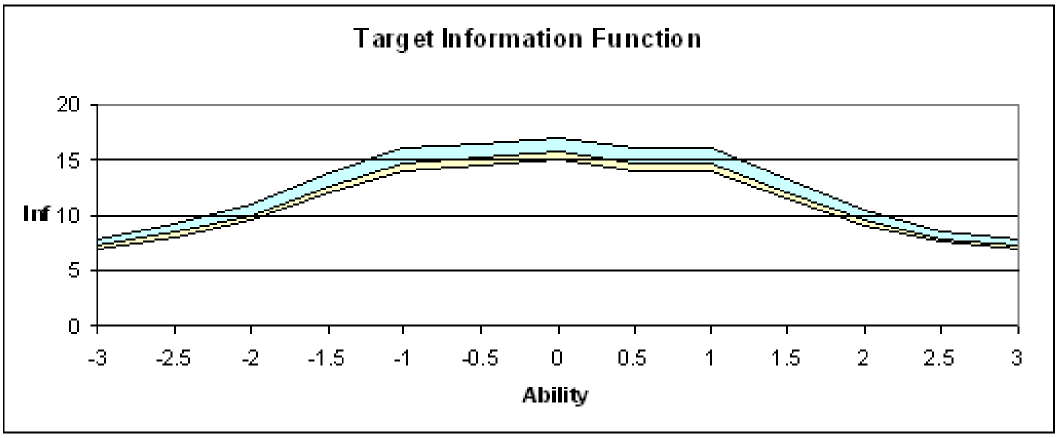

Figure 1. Example of a Target Information function.

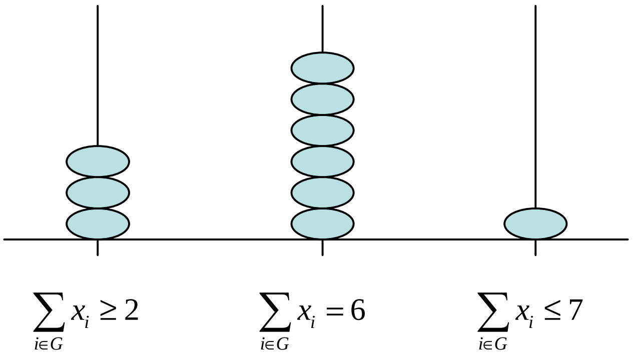

Figure 2. Application of marble valorization procedure

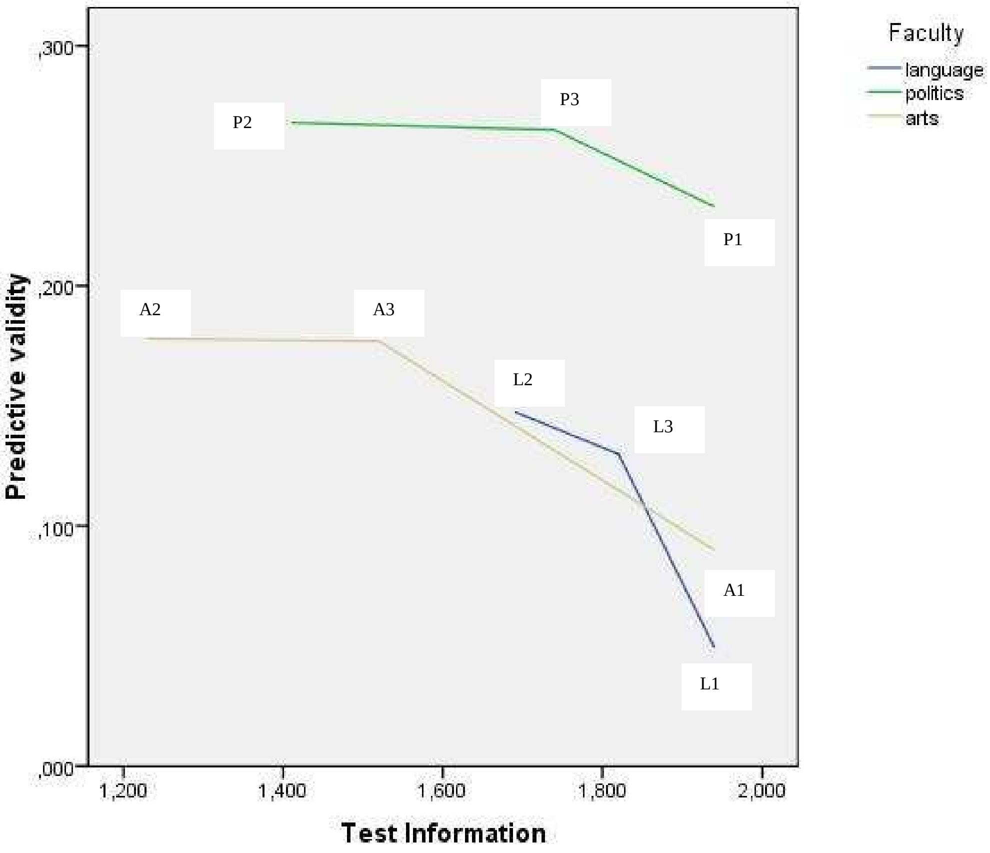

Figure 3. Results of different test assembly models 1, 2 and 3 for Language (L), Politics (P) and Arts (A)