Available online at www.sciencedirect.com

ScienceDirect

European Journal of Operational Research 193 (2009) 342–350

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/ejor

Discrete Optimization

A mixed integer programming model for multiple stage adaptive testing ☆

Jennifer Edmonds [a,*], Ronald Armstrong [b]

[a] *Sidhu School of Business & Leadership, Wilkes University, 84 W. South Street, Wilkes-Barre, PA 18766, USA*
[b] *Rutgers Business School, Rutgers University, Newark/New Brunswick, NJ, USA*

Received 24 October 2006; accepted 27 October 2007
Available online 5 November 2007

Abstract

The last decade has seen paper-and-pencil (P&P) tests being replaced by computerized adaptive tests (CATs) within many testing programs. A CAT may yield several advantages relative to a conventional P&P test. A CAT can determine the questions or test items to administer, allowing each test form to be tailored to a test taker's skill level. Subsequent items can be chosen to match the capability of the test taker. By adapting to a test taker's ability, a CAT can acquire more information about a test taker while administering fewer items. A Multiple Stage Adaptive test (MST) provides a means to implement a CAT that allows review before the administration. The MST format is a hybrid between the conventional P&P and CAT formats. This paper presents mixed integer programming models for MST assembly problems. Computational results with commercial optimization software will be given and advantages of the models evaluated.
Published by Elsevier B.V.

Keywords: Integer programming; Linear programming; Education

1. Introduction

Traditional paper and pencil (P&P) tests are linear tests and the items or questions are fixed for every examinee during an administration. In computer adaptive testing (CAT), examinees receive items that match their current ability estimate. The last decade has seen paper-and-pencil (P&P) tests being replaced by computerized adaptive tests (CATs) within many testing programs. A multiple stage computer adaptive test (MST) combines characteristics of both a standard CAT and P&P test because it adapts to the ability of the examinee like CAT and provides P&P benefits such as test specialist review, exposure of preselected items and parallel test forms.

An MST is an ordered collection of testlets (Wainer and Kiely, 1987; Luecht and Nungester, 1998) that allows for adaptation based on a test taker's ability while exposing a pre-set number of items. A testlet is a group of items that is developed as a unit. This test structure is a hybrid between the conventional paper-and-pencil (P&P) test and a CAT. An MST adapts over testlets instead of over individual items as in traditional computer adaptive testing; a test-taker's progression through the network of testlets adapts to the test-taker's ability. It is a computerized extension of the early attempts at adaptive tests by Lord (1971a,b), where items given at later stages depended on performance in earlier stages.

The MST structure contains multiple stages and consists of *bins* in which testlets are placed. The bins at a given stage are arranged in levels according to ability classifications. Most of the MST designs have two testlets at the beginning, where the routing of the test-taker from one testlet to the next is automatic. The collection of paths through the network yields the set of all possible test forms. The

☆ This research has been funded by a grant from the Law School Admission Council.

* Corresponding author. Tel.: +1 570 4084719; fax: +1 540 4087874.
E-mail address: jennifer.edmonds@wilkes.edu (J. Edmonds).

0377-2217/$ - see front matter Published by Elsevier B.V.
doi:10.1016/j.ejor.2007.10.047

following figure shows a diagram of a possible design for a multiple stage adaptive test with four possible paths or test forms.

A Multiple Stage Adaptive Test Design (MSTD) is a framework for a class of MSTs; that is, an MSTD that has no items or testlets associated with it. An MSTD gives the position of bins designed for a test taker classification, and a testlet is placed in each bin to create an MST. The sequence of bins a test taker may follow yields the paths through the MST. The combined testlets on a path yields a test form of the MST. The MSTD also states the constraints for every path. This paper introduces models to assemble multiple MSTs from an MSTD. All MSTs assembled for an MSTD are considered parallel to one another, as standardized linear test forms are considered parallel to each other. The primary objective is to obtain as many parallel test forms from the item pool as possible.

The use of mathematical programming techniques to assemble test forms is common at testing agencies. These techniques save thousands of hours of personnel time. A test form assembled by a computer can be assured to satisfy all stated test specifications. While the review of the form by test specialists may still be desirable, the review generally entails a small number of alterations to account for constraints not coded in the database. Notable heuristic methods, also referred to as sequential item selection, for test assembly are Ackerman (1989), Adema (1990), Luecht and Hirsch (1992), Luecht (1998), and Swanson and Stocking (1993). These heuristics generally provide a viable approach to automatic test assembly, but in situations where a small feasible region exists, a test satisfying all the constraints may not be found.

Boekkooi-Timminga (1987) used 0–1 linear programming for simultaneous test assembly. These models are very flexible as most test assembly problems can be modeled this way. However, these searches sometimes require a large amount of computer processing time. The complexity of these problems increases with larger item pools, an increased number of variables and constraints. Theunissen (1985), Boekkooi-Timminga (1987), Adema (1990), van der Linden (1998), and van der Linden (2000) proposed the use of a more general mixed integer programming (Nemhauser and Wolsey, 1988) software package. This paper takes the last approach and uses a commercially available code, CPLEX (ILOG, 2002), as the tool to obtain solutions to mixed integer programming (MIP) problems arising from the MST assembly models. The MIP problems are constructed with the modeling language AMPL (Fourer et al., 2003). The process to be presented could be implemented with any software package that can solve large-scale mixed integer programming problems. Formulated as a single integer programming problem, the problem of obtaining the most feasible MSTs from an item pool has over 250,000 binary variables and about 400,000 constraints for a relative small pool. Decomposition techniques (Dantzig and Wolfe, 1960) can be used to solve the problem optimally. These techniques are difficult to implement and generally converge slowly. This paper utilizes a compromise between obtaining tests one at a time and solving the problem by attempting to solve a single MIP.

Although the approach of this paper can easily be extended to the situation where several items relate to a single stimulus, this paper limits itself to models for discrete items where an item's stimulus and question can be treated as a unit. The next section gives an example of an MSTD and demonstrates how it can be represented with a tree structure. The following section presents a generic model, and then modifications of the model are given. These modifications include additional practical constraints and adjustments to facilitate the solution with a commercial MIP package. A shadow test approach for assembling multiple MSTs is outlined. Computational results are given to compare model variations, and shadow test assembly approaches versus sequential assembly of multiple MSTs. The concluding section discusses how the assembly might take place in an operational setting.

1.1. Example of an MSTD

MSTDs differ in the number of bins, number of items per bin, the number of stages, the target curves and the constraints. All MSTs corresponding to a particular MSTD will be similar in all of these attributes. Table 1 depicts an example of an MSTD that is being evaluated. The layout depicts an MSTD having three stages with a total of six bins. The bins at a given stage are arranged in levels corresponding to item difficulty/test taker ability classifications or strata. In this example, every MST bin will be assigned two testlets with five, six or seven items. Every possible path through the MST constitutes a test form, and each path has between 35 and 37 items. The path that a particular test taker may traverse through an MST obtained from this design contains exactly two testlets from each stage. More flexible designs can be developed where the administration can be terminated based on the confidence interval of an ability estimate; but this is not considered here.

The convention used for numbering the bins is sequential starting at the first stage. The numbering within a stage

Table 1
An MST design with 3 stages and 6 bins is given

Stage Percentile	(i) 10–14 items	(ii) 10–14 items	(iii) 10–14 items
[67, 100]			Bin 6
[50, 100]		Bin 3	
[0, 100],[33, 67]	Bin 1		Bin 5
[0, 50]		Bin 2	
[0, 33]			Bin 4

Each bin in the MST depicted is targeted for a particular population percentile range as indicated in the left margin. The range on the number of items assigned to each bin is given in the column header. This design has two testlets in each bin and 5, 6 or 7 items in every testlet.

دریافت فوری ← متن کامل مقاله

ISIArticles
مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات