A Study of a Network-Flow Algorithm and a Noncorrecting Algorithm for Test Assembly

R. D. Armstrong, D. H. Jones, and Xuan Li

Rutgers, The State University of New Jersey

Ing-Long Wu, National Yulin Institute of Technology, Taiwan

The network-flow algorithm (NFA) of Armstrong, Jones, & Wu (1992) and the average growth approximation algorithm (AGAA) of Luecht & Hirsch (1992) were evaluated as methods for automated test assembly. The algorithms were used on ACT and ASVAB item banks, with and without error in the item parameters. Both algorithms matched a target test information function on the ACT item bank, both before and after error was introduced. The NFA matched the target on the ASVAB item bank; however, the AGAA did not, even without error in this item bank. The AGAA is a noncorrecting algorithm, and it made poor item selections early in the search process when using the ASVAB item bank. The NFA corrects for nonoptimal choices with a simplex search. The results indicate that reasonable error in item parameters is not harmful for test assembly using the NFA or AGAA on certain types of item banks. *Index terms: algorithmic test construction, automated test assembly, greedy algorithm, heuristic algorithm, item response theory, marginal maximum likelihood, mathematical programming, simulation, test construction.*

Birnbaum (1968) proposed maximizing Fisher's information to create a test based on item response theory (IRT). Lord (1980, p. 23) presented iterative steps for a heuristic method to match a set of predetermined information values at specific trait levels. Recently, research on automated test construction has focused on three categories of algorithmic approaches. The first category involves the development of mathematical programming models with formal constraints and solution techniques based on a simplex search (Armstrong, Jones, & Wu, 1992; Baker, Cohen, & Barmish, 1988; Boekkooi-Timminga, 1987, 1990; van der Linden & Boekkooi-Timminga, 1989; Theunissen, 1985, 1986). The second category consists of noncorrecting approaches that may not formally search for a constrained-optimal solution (e.g., Ackerman, 1989; Luecht & Hirsch, 1992). A third category is a greedy heuristic followed by an interchange heuristic, or self-correcting steepest descent approach (e.g., Swanson & Stocking, 1993).

Noncorrecting algorithms, although fast, can become "trapped" into searching suboptimal paths in the solution space and may arrive at a worthless solution (Nemhauser & Wolsey, 1988). Network-flow algorithms find a global optimum in a reasonable time; however, optimal solutions may be highly sensitive to small changes in the parameters defining the solution space. The objective of this study was to evaluate the quality of tests generated by a noncorrecting algorithm versus a network-flow algorithm, and to study the sensitivity of the solutions to item bank error. This study used the network-flow algorithm (NFA) of Armstrong et al. (1992) and the noncorrecting average growth approximation algorithm (AGAA) of Luecht & Hirsch (1992) because these two competing algorithms are fast enough to make automated test assembly practical.

Item Response Theory

A test consists of m items and is selected from an item bank containing n items. Denote the response to

0146-6216/96/010089-10$1.75

item i as u_i. For dichotomous items, u_i = 1 denotes a correct response and u_i = 0 denotes an incorrect response. Various models for the probability of correct response are available (Lord, 1980). The three-parameter logistic model was used here. For trait level θ (a real number)

$$P(\theta;a_i,b_i,c_i)=c_i+(1-c_i)\frac{\exp(\mathrm{D}z_i)}{1+\exp(\mathrm{D}z_i)}, \tag{1}$$

where

a_i is the discrimination parameter for item i,
b_i is the difficulty parameter for item i,
c_i is the pseudoguessing parameter for item i,
z_i $= a_i(\theta - b_i)$, and
D is a scaling factor equal to 1.7.

Test Information

The primary criterion (Lord, 1980) for constructing tests in IRT is the test information function (TIF). The item information function is Fisher's information and is given by

$$I(\theta;a_i,b_i,c_i)=-\mathrm{E}_u\left\{\frac{\partial^2\log f(u;\theta,a_i,b_i,c_i)}{\partial^2\theta}\right\}=\frac{[\partial P(\theta;a_i,b_i,c_i)/\partial\theta]^2}{P(\theta;a_i,b_i,c_i)[1-P(\theta;a_i,b_i,c_i)]}. \tag{2}$$

The TIF is

$$I(\theta;\Phi)=\sum_{i=1}^{m} I(\theta;a_i,b_i,c_i). \tag{3}$$

A Measure of Weak Parallelism

Under regularity assumptions, TIFs measure the precision of a test along θ. Different tests may require different levels of precision along θ. Two or more tests are weakly parallel if they have identical TIFs and they measure the same trait (Lord, 1977; Samejima, 1977). Test constructors may specify a target test information function (TTIF). It is also possible that the test constructor might have a reference test form from which to construct the TTIF. This reference test becomes the target test, and the information function of a new test must fit the TTIF.

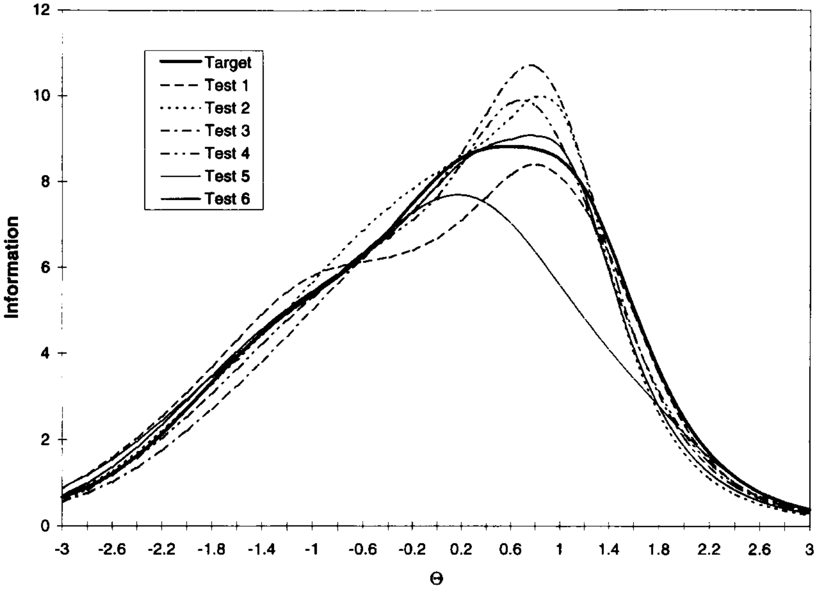
A coefficient of goodness of fit, R^2, is proposed to measure the degree of weak parallelism between a target test and a test generated by an algorithm. Let $T(\theta)$ denote a TTIF at θ, and let $\overline{T} = \int T(\theta)h(\theta)\mathrm{d}(\theta)$ be the mean value, where $h(\theta)$ is the density of the examinee population. Let $A(\theta)$ denote the information function of an algorithmic test—in other words a test generated by a computer algorithm.

A normalized squared difference, analogous to the coefficient of determination used in least-squares regression, is

$$R^2=1-\frac{\int[T(\theta)-A(\theta)]^2h(\theta)d\theta}{\int[T(\theta)-\overline{T}]^2h(\theta)d\theta}. \tag{4}$$

In general, $A(\theta)$ is not an orthogonal projection of $T(\theta)$ and R^2 may be negative. If R^2 = 1, the algorithmic information function is weakly parallel to the TTIF. Figure 1 shows seven TIFs, one target and six algorithmic tests, with R^2 = .81, .94, .81, .95, .76, and .97 for Tests 1–6, respectively. Visual inspection suggests that the TIFs of the six tests are close to the target test (i.e., each algorithmic test could be considered weakly parallel to the target test).

Figure 1
TIFs of Six Generated Tests and One Target Test Based on the ASVAB Item Bank ($\overline{R}^2$ = .87)



Two Test-Generation Algorithms

Two algorithms were used to generate parallel tests—the NFA and the AGAA. The objective of test generation algorithms is to maintain the characteristics of a target test in each algorithmic test. The item bank includes the *a*, *b*, and *c* parameters, along with a content identifier, for each item. The target test data contain the IRT parameters for each item and summary requirements on the content.

The Network-Flow Item Matching Algorithm (NFA)

Gulliksen (1950) originated the item matching idea to divide a test into two parallel tests to estimate test reliability. van der Linden & Boekkooi-Timminga (1988) proposed a zero-one programming model to perform item matching when constructing parallel tests. The network-flow algorithm (NFA) of Armstrong et al. (1992) was studied here. The NFA item matching technique uses any metric of the distance between two items, and assumes the existence of a target test.

Interitem Distances

Interitem distance measures are useful in test assembly algorithms; two such distances are the Euclidean and L_p distance measures (see Pearson, 1974, pp. 498–515). Suppose item *i* and item *t* are compared. The Euclidean distance is

$$d'_{it} = \left[w_a(a_i - a_t)^2 + w_b(b_i - b_t)^2 + w_c(c_i - c_t)^2\right]^{1/2}, \tag{5}$$

where (a_i, b_i, c_i) is the item parameter vector for item *i* in the item bank, and (a_t, b_t, c_t) is the item parameter vector of item *t* in the target test. In addition, $w_a \geq 0$, $w_b \geq 0$, $w_c \geq 0$, and $w_a + w_b + w_c = 1$.

The L_p distance is

$$d''_{it} = \left[\int \left|I(\theta; a_i, b_i, c_i) - I(\theta; a_t, b_t, c_t)\right|^p h(\theta) d(\theta)\right]^{1/p}, \quad p \geq 1. \tag{6}$$

Implementing the NFA

The NFA objective function uses a sum of the interitem distances, such as the Euclidean distance (Equation 5) or the L_p (Equation 6), rather than a direct measure of distance between two TIFs, such as R^2 (Equation 4). Thus, the NFA fits the target test on an item-by-item basis. The resulting tests are approximately weakly parallel. For the L_p distance, Minkowski's inequality (Pearson, 1974, p. 516) can be used to show that the sum of interitem distances is an upper bound to the distance between TIFs. For the Euclidean distance, test parallelism follows because item information is a continuous function of the item parameters, and when the item parameters of two tests are approximately equal the TIF of the tests will be approximately equal.

In Phase I, items needed for K tests are selected to match the target test by minimizing an appropriate distance function and satisfying all content requirements. The mathematical programming problem resulting from this model can be solved optimally using the NFA. Using only the items selected in Phase I, K parallel tests are created in Phase II. Algorithmic tests are balanced for the Euclidean distance or the L_p distance while satisfying the content constraints on each test. The Phase II problem is solved with an interchange heuristic. Armstrong et al. (1992) contains a more detailed model description and discussion of solution methods.

Phase I. The first phase selects a subset of items from the item bank to optimally match the target test. The mathematical model is: Minimize

$$Z_1 = \sum_{i=1}^{n} \sum_{t=1}^{m} d_{it} x_{it} , \tag{7}$$

subject to

$$\sum_{t=1}^{m} x_{it} + y_i = 1, \quad i = 1, \ldots, n, \tag{8}$$

$$\sum_{i=1}^{n} x_{it} = K, \quad t = 1, \ldots, m, \tag{9}$$

$$\sum_{i \in C_g} y_i = n_g - K \times c_g, \quad g = 1, \ldots, G, \tag{10}$$

and

$$x_{it}, y_i \in \{0,1\}, \tag{11}$$

where

x_{it} = 1 if item i of the item bank is matched to item t in the target test and 0 otherwise;
y_i = 1 if item i is assigned to the group of unused items;
d_{it} is the distance between item i in the item bank and item t in the target test (either the Euclidian distance or L_p);
K is the number of algorithmic tests;
n is the number of items in the item bank;
m is the number of items in each target test;
G is the number of content groups;
C_g is the set of all items with the same content g in the item bank;
n_g is the number of items in the pool in set C_g; and
c_g is the number of items from content group g required in a test.

The objective function in Equation 7 minimizes the sum of the distance between all selected items in the item bank and the matched target items. The constraints in Equation 8 assign each item in the item bank to either the target or the group of unused items. The constraints in Equation 9 assign K items to each target item to generate K parallel tests. The constraints in Equation 10 guarantee that the number of unused items for content group g will be $N_g - K \times c_g$ because Phase I has to allocate $K \times c_g$ items for the test forms in content group g. The constraints in Equation 11 are the integer restrictions.

Phase II. Phase II generates K tests from the pool of items selected in Phase I. The model balances the deviation of the individual test from the target test and enforces the content requirements. The mathematical model is: Minimize Z_2, subject to

$$\sum_{t=1}^{m} \sum_{i \in D(t)} d_{it} u_{itK} \le Z_2, \quad k = 1, \ldots, K, \tag{12}$$

$$\sum_{k=1}^{K} u_{itk} = 1, \quad i \in D(t), \tag{13}$$

$$\sum_{i \in D(t)} u_{itk} = 1; \quad t = 1, \ldots, m; \quad k = 1, \ldots, K, \tag{14}$$

$$\sum_{t=1}^{m} \sum_{i \in D(t) \cap C_g} u_{itk} = c_g; \quad g = 1, \ldots, G; \quad k = 1, \ldots, K, \tag{15}$$

and

$$u_{ikt} \in \{0,1\}. \tag{16}$$

The set $D(t)$ is defined to be the collection of item indexes matched with target item t by Phase I. The number of items in $D(t)$ will be K. The decision variable, u_{itk}, is equal to 1 if item i in $D(t)$ is matched to target item t and assigned to the parallel test k, and 0 otherwise.

The objective function combined with the constraints in Equations 12–16 achieves the balance of the total interitem distance from the target test. If the optimal objective function from Phase I is Z_1*, then the target solution has an objective value of Z_1*/K. The constraints in Equation 13 force each item matched to a target item in Phase I to be assigned to exactly one test. The constraints in Equation 14 ensure that each target item on each test has an item from the bank assigned to it. The constraints in Equation 15 guarantee that content requirements are satisfied for each test.

The Average Growth Approximation Algorithm (AGAA)

Luecht & Hirsch (1992) presented an algorithm using an average growth approximation of the TTIF. The AGAA builds a test by adding items successively to satisfy an average goal. Experience with the AGAA indicates that it is as fast as the NFA. However, the AGAA does not set out to solve a mathematical programming model with integer constraints. The AGAA determines its objective within content areas using TTIFs within each content area.

The AGAA selects the best item to match the TTIF (divided by the number of test items). The difference between the TTIF and the algorithmic TIF is computed at selected points using an appropriate distance function. The differences are divided by the remaining number of items that the test still requires. The item information function of the available items is also calculated at these points. The sums of the weighted absolute deviations of each available item's information function value and the test's average information requirement are computed. The item with the smallest absolute deviation is selected from the item bank for the test. The process continues until the test is complete.

The algorithm is implemented in two stages. The first stage fits the information functions of the subtests. The second stage combines the composite grouping of items for the subtests to fit the overall TTIF. The AGAA does not preclude item-by-item matching. In general form, the AGAA selects items sequentially to minimize

$$S_i = \int \left| \frac{T(\theta) - T^*(\theta)}{m - m^* + 1} - I(\theta; a_i, b_i, c_i) \right| f(\theta) d\theta, \qquad (17)$$

where

$T(\theta)$ is the target information function,
$T^*(\theta)$ is the information function for the test under construction,
m^* is the number of selected items, and
$f(\theta)$ is a proportional weight denoting the unclaimed information in the target information function.

Changing $T(\theta)$ to $I(\theta; a_i, b_i, c_i)$ and letting $T^*(\theta) = 0$ is a manipulation that allows the AGAA to match the target test item-by-item. Stated in this fashion, the AGAA is functionally similar to an L_1 distance, or absolute value distance, related to Equation 6.

Simulation Comparison of NFA and AGAA

Method

Simulating Marginal Maximum Likelihood Estimates of the Item Parameters

Marginal maximum likelihood estimates (MMLEs) of item parameters were simulated using a procedure that is asymptotically equivalent to estimating item parameters based on monte carlo generated data. This procedure first creates the variance-covariance matrix of the item parameters based on N examinees. Then, item parameters are simulated from the variance-covariance matrix and the assumed-true item parameters. This approach, based on the asymptotic normality of the sampling distribution of the MMLEs, is faster than the standard simulation method.

The simulation has as input the $3m \times 1$ array of the true item parameters $\Phi_0 = \{\phi_{l0}: l = 1, 2, ..., 3m\}$ and the $3m \times 3m$ covariance matrix $[N\mathbf{A}'\mathbf{A}]^{-1}$, where N is the number of examinees, m is the number of items in a test, and the elements of the $3m \times 3m$ Fisher information matrix, $\mathbf{A}'\mathbf{A}$, are:

$$\sum_{\mathbf{u}\in\mathbf{U}} \left\{ \frac{\partial \log[\pi_{\mathbf{u}}(\Phi_0)]}{\partial \phi_l} \right\} \left\{ \frac{\partial \log[\pi_{\mathbf{u}}(\Phi_0)]}{\partial \phi_{l'}} \right\} \pi_{\mathbf{u}}(\Phi_0). \qquad (18)$$

The procedure generates multivariate normal random variables with the variance-covariance matrix $[N\mathbf{A}'\mathbf{A}]^{-1} = \mathbf{cc}'$, where $\mathbf{c}$ is a $3m \times 3m$ lower triangular matrix. The procedure uses two steps:

1. Generate v_l, $l = 1, 2, ..., 3m$, iid N(0,1) random variables.
2. For $l = 1, 2, ..., 3m$; let

$$\hat{\phi}_l = \phi_{l0} + \sum_{l'=1}^{l} c_{ll'} z_{l'} . \qquad (19)$$

Let $\mathbf{u} = \{\hat{\phi}_l: l = 1, 2, ..., 3m\}$ be the array of simulated maximum likelihood estimastes (MLEs). The number of examinees, N, determines the degree of error.

Data

The objective of this study was to measure the fit between the algorithmic tests and the target test using

R^2. The items for the algorithmic tests were selected from the simulated item bank using either the NFA or the AGAA with simulated item parameters based on ASVAB and ACT item banks.

The ASVAB item bank consisted of 450 items on the Arithmetic Reasoning subtest from the ASVAB (Shore, 1989) distributed over four content areas (see Table 1). The target test consisted of 30 items distributed over the corresponding content areas with 10, 11, 4, and 5 items in each content area.

Table 1
Item Parameter Means and Standard Deviations (SDs) of the ASVAB and ACT for Each Content Group and Overall

Test, Content Group,	a		b		c	
and Number of Items	Mean	SD	Mean	SD	Mean	SD
ASVAB						
1, $n = 150$	1.079	.409	−.467	1.179	.210	.095
2, $n = 165$	1.128	.438	−.154	1.033	.200	.104
3, $n = 60$	1.092	.532	−.025	.815	.203	.084
4, $n = 75$	1.237	.383	−.014	.678	.162	.080
All, $n = 450$	1.125	.414	−.218	1.024	.197	.096
ACT						
1, $n = 48$	1.015	.292	−.485	.465	.154	.044
2, $n = 168$	.911	.322	.131	.977	.160	.054
3, $n = 24$	1.028	.328	.811	.778	.173	.059
4, $n = 96$	1.120	.419	.689	.655	.167	.058
5, $n = 96$	1.037	.356	.527	.650	.151	.062
6, $n = 48$	.911	.312	.475	.828	.163	.058
All, $n = 480$	.994	.355	.329	.864	.160	.056

The ACT item bank consisted of 480 items on 12 previously administered mathematics tests from the American College Testing Assessment Test (Luecht, 1991) distributed over six content areas (see Table 1). The target test consisted of 40 items distributed over the corresponding content areas with 14, 4, 8, 8, 4, and 2 items in each content area.

The 480 ACT items were a compilation of 12 parallel test forms of 40 items each. Item parameters were simulated as if they had been calibrated with these 12 tests. The ASVAB item bank originally consisted of 576 pretest items. Because the ASVAB does not have a natural partitioning into test forms, it was broken into 15 artificial, parallel "tests" of 30 items, and the remaining items were discarded for this study. Item parameters were simulated as if they had been calibrated with these 15 "tests."

The simulated MLEs of the item parameters were generated using four sample sizes (N = 500, 1,000, 1,500 and 2,000). 1,000 simulated item banks were created for each value of N. The study involved two assumed-true item banks; thus, 8,000 simulated item banks were created, which when combined with the original ASVAB and ACT item banks yielded 8,002 item banks.

Analysis

Six parallel tests were generated from the assumed-true item bank and every simulated item bank. Two Euclidean distance measures (d'_{ij}) and the L_p distance (d''_{ij}) were used for the NFA along with the AGAA; thus, 8,002 × 6 × 4 tests were generated. The Euclidean-A distance measure had $w_a = .20$, $w_b = .75$, and $w_c = .05$; the Euclidean-B distance measure had $w_a = .25$, $w_b = .75$, and $w_c = 0$. The selection of the values for these weights was discussed by Armstrong et al. (1992).

The average R^2, $\overline{R}^2$, was computed over six algorithmic tests from each simulation of the item bank. It is possible for tests generated from the simulated item banks to have an $\overline{R}^2$ value higher than this coefficient because the $\overline{R}^2$ was not the criterion being maximized. Note that $\overline{R}^2$ does not measure the performance of an algorithm with its own criterion. However, $\overline{R}^2$ is comparable for different types of algorithms. Because neither the NFA nor the AGAA use $\overline{R}^2$ for their criterion, this measure does not favor one approach over another.

Results

Table 2 shows the results for the NFA and AGAA on each item bank for each value of N. (The value of $\overline{R}^2$ when tests were generated from the true item bank is found in the row denoted $N = \infty$.) In most cases, the distribution of $\overline{R}^2$ increased stochastically as N increased and the MLEs approached their true parameter values, as expected. The largest increase in R^2 usually occurred with an increase from $N = 500$ to $N = 1{,}000$; for example, for the L_p measure and 50th percentile, R^2 increased from .83 to .91. Using Figure 1 with $\overline{R}^2 = .87$ as a benchmark, these algorithmically constructed tests would compare well with tests using test construction experts. The NFA did well on both item banks with the R^2 50th percentile no lower than .76, but the AGAA did well only on the ACT item bank with a 50th percentile no greater than .42 on the ASVAB.

Table 2
The 25th, 50th, and 75th Percentiles of $\overline{R}^2$ for the ASVAB and ACT as a Function of Sample Size (N) and Type of Algorithm

	NFA											
Test	L_p			Euclidean-A			Euclidean-B			AGAA		
and N	25	50	75	25	50	75	25	50	75	25	50	75
ASVAB												
$N = 500$	.80	.83	.87	.77	.81	.85	.72	.76	.81	.29	.39	.49
$N = 1{,}000$	.90	.91	.93	.87	.89	.90	.82	.85	.87	.32	.41	.50
$N = 1{,}500$	.93	.94	.95	.89	.91	.93	.85	.87	.91	.33	.41	.49
$N = 2{,}000$	.94	.95	.96	.90	.92	.93	.87	.89	.91	.32	.42	.49
$N = \infty$	.97	.97	.97	.93	.93	.93	.94	.94	.94	.36	.36	.36
ACT												
$N = 500$	.89	.90	.93	.88	.90	.92	.88	.89	.92	.87	.88	.91
$N = 1{,}000$	.95	.95	.96	.94	.95	.95	.94	.95	.95	.91	.92	.93
$N = 1{,}500$	.96	.97	.98	.96	.96	.97	.96	.96	.97	.93	.94	.95
$N = 2{,}000$	.97	.98	.98	.97	.97	.97	.97	.97	.97	.94	.95	.95
$N = \infty$	.98	.98	.98	.98	.98	.98	.98	.98	.98	.94	.94	.94

Of the three interitem distances, L_p was the best for the NFA with the highest R^2 overall. This may be because L_p directly measures the item information function across θ, and the other distances do not. The Euclidean-A results were better than those of Euclidean-B. This may be because $w_c = 0$ in Euclidean-B. No attempt was made to optimize parameter weights, but in other computational work (see Wu, 1992) the weights used here were found to be acceptable.

The AGAA performed well on the ACT item bank, but did not do well on the ASVAB item bank. For example, the 50th percentile of R^2 in Table 2 is approximately .40 for all values of N. As a plausible explanation, note that for the results with no item parameter errors ($N = \infty$), the (nonstochastic) value of $\overline{R}^2$ was .94 for the ACT versus .36 for the ASVAB; thus, the difference in the performance of AGAA for the ASVAB and the ACT item banks was not due to errors in the item parameters.

Discussion and Conclusions

$\overline{R}^2$ measures differences in information functions using weights from the standard normal probability mass function h. Although it is possible that this measure was detrimental to the AGAA on the ASVAB because the AGAA does not use h to build tests, this is unlikely because the $\bar{R}^2$ was high (>.80) for AGAA on the ACT item bank; this would not be expected if h was causing the poor performance of the AGAA on the ASVAB. In addition, the Euclidean-A and Euclidean-B interitem distances do not use h, but they had high $\bar{R}^2$ values for both item banks.

The failure of AGAA with the ASVAB item bank might be due to its being a member of the "greedy" class of methods (Nemhauser & Wolsey, 1988). The main feature of a greedy heuristic is that it selects an item performing locally best according to an appropriate criterion at each iteration, and retains it until the end of the procedure. In other words, at a given stage of the iterative process, a greedy heuristic does not have the option to "swap-out" a particular item in favor of another. This keeps it from "jumping" to a totally different part of the solution space. This feature puts a greedy method at risk of finding a substantially suboptimal solution if poor item choices are made early in the search. The ASVAB item bank might have provided the AGAA with more divergent path choices because of the larger variation in the item parameters. The AGAA could possibly be improved by some form of interchange heuristic.

The results of this study showed that the calibration of item parameters with N as low as 2,000 did not seriously degrade the performance of the NFA or AGAA on the ACT item bank and that of the NFA on the ASVAB item bank. The performance of the AGAA on the ASVAB appeared to be unrelated to error in the item parameters. Both algorithms are recommended, with reservations, for generating tests with simple content requirements such as those studied here. The AGAA failed for the ASVAB item bank. Thus, the AGAA should be used with caution. Practitioners should examine benchmark measures such as R^2 to judge the quality of an algorithmic test, so they would know when the AGAA has failed.

References

Ackerman, T. A. (1989, March). *An alternative methodology for creating parallel test forms using the IRT information function.* Paper presented at the annual meeting of the National Council on Measurement in Education, San Francisco.

Armstrong, R. D., Jones, D. H., & Wu, I.-L. (1992). An automated procedure for test development of tests parallel to a seed test. *Psychometrika, 57,* 256–271.

Baker, F. B., Cohen, A. S., & Barmish, B. R. (1988). Item characteristics of tests constructed by linear programming. *Applied Psychological Measurement, 12,* 189–199.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. M. Lord & M. R. Novick, *Statistical theories of mental test scores* (pp. 397–479). Reading MA: Addison-Wesley.

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.

Gulliksen, H. (1950). *Theory of mental tests.* New York: Wiley.

Lord, F. M. (1977). Practical applications of item characteristic curve theory. *Journal of Educational Measurement, 14,* 117–138.

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

Luecht, R. M. (1991). [American College Testing Program: Experimental item pool parameters.] Unpublished raw data.

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

Nemhauser, G., & Wolsey, L. (1988). *Integer and combinatorial optimization.* New York: Wiley.

Pearson, C. E. (1974). *Handbook of applied mathematics.* New York: van Nostrand Reinhold.

Samejima, F. (1977). Weakly parallel tests in latent trait theory with some criticisms of classical test theory. *Psychometrika, 50,* 411–420.

Shore, W. (1989). [Armed Services Vocational Aptitude Battery: Experimental item pool parameters.] Unpublished raw data.

Swanson, L., & Stocking, M. L. (1993). A model and heuristic for solving very large item selection problems. *Applied Psychological Measurement, 17,* 151–166.

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

Theunissen, T. J. J. M. (1986). Some applications of optimization algorithms in test design and adaptive testing. *Applied Psychological Measurement, 10,* 381–389.

van der Linden, W. J., & Boekkooi-Timminga, E. (1988). A zero-one programming approach to Gulliksen's matched random subsets. *Applied Psychological Measurement, 12,* 201–209.

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

Wu, I. L. (1992). *Matching procedures to create standardized parallel tests.* Unpublished doctoral dissertation, Graduate School of Management, Newark NJ.

Acknowledgments

The work of D. H. Jones was supported by the Navy Manpower, Personnel, and Training R&D Program of the Office of the Chief of Naval Research under Contract N00014-87-C-0696. The authors acknowledge the advice and consultation given by Zhaobo Wang.

Author's Address

Send requests for reprints or further information to Douglas H. Jones, Faculty of Management, Rutgers The State University, 180 University Avenue, Newark NJ 07102, U.S.A. Email: dhjones@rci.rutgers.edu.