Article

Software Review of flexMIRT Version 3.5

Applied Psychological Measurement
1–16

DOI: 10.1177/0146621617726792
journals.sagepub.com/home/apm
SAGE

S. Austin Cavanaugh[1], Ciji A. Heiser[1], Karen B. Hoeve[1], Eren Halil Ozberk[2], Elizabeth A. Patton[1], John C. Sessoms[1], Myrah R. Stockdale[3], Elif Bengi Unsal-Ozberk[4], and Claire Wood[1]

Abstract

flexMIRT is a versatile program for unidimensional and multidimensional item response theory (IRT) calibrations, scoring analyses, and model-based simulations. With an adaptable syntax that allows for various combinations of model specifications, estimation constraints, and estimation choices, flexMIRT can handle almost all of the most popular IRT models for dichotomous and polytomous data. The software package also supports diagnostic classification models and multigroup and multilevel analyses. This review evaluates the software from a user's perspective as well as some of its calibration, scoring, and simulation capabilities. Two simulation studies are included: one demonstrates some basic simulation capabilities and the other provides some direct comparisons with BILOG-MG. The review suggests that flexMIRT is a very good product that is only likely to get better as new features and suggestions for improvement are implemented.

Keywords

item response theory, multilevel models, estimation, flexMIRT, BILOG-MG, polytomous models

Introduction

flexMIRT (Cai, 2017) is an item response theory (IRT) calibration and scoring software package that handles almost any combination of unidimensional or multidimensional models for dichotomous and polytomous data, with support for multilevel and multi-group analyses, as well as integrated support for graphics using the R programming language. The package is licensed and distributed by Vector Psychometric Group (VPG), LLC (www.vpgcentral.com). There are a variety of licensing options available, including significantly discounted options for faculty and students.

This review covers four important aspects of use: (a) software installation and usability, (b) calibration and test scoring capabilities, (c) simulation and parameter recovery analyses, and (d) comparisons with the BILOG-MG calibration and test scoring software package. An in-depth

[1]University of North Carolina at Greensboro, USA
[2]Hacettepe University, Ankara, Turkey
[3]Campbell University, Buies Creek, NC, USA
[4]Ankara University, Turkey

Corresponding Author:
S Austin Cavanaugh, University of North Carolina at Greensboro, P.O. Box 26170, Greensboro, NC 27402, USA.
Email: sacavana@uncg.edu

analysis of all of the software features and options is obviously beyond the scope of this review. Some hopefully pragmatic features and suggestions are made for VPG to consider for future versions of the software.

Installation and Usability

flexMIRT currently only runs on Microsoft Windows® Version 7 or later (or operating systems with compatible emulation) with available 32-bit and 64-bit versions. The program installation folder includes the executable application file, a 237-page user manual, and syntax template files; it occupies less than 7 megabytes of disk space. The entire installation process is handled online.

Users must create an online account with VPG and accept the licensing agreement. Up to three installations are allowed with a purchased license. The authors found VPG to be very responsive in addressing any software installation or licensing issues.

The end-user must have system administrator rights on the installation computer. Once downloaded, the application installs very quickly—usually in less than a few minutes, depending on the quality of the Internet connection. The first time the flexMIRT application is opened, users must formally register the software using their login credentials or their license code. Some users may re-experience the registration pop up each time they open flexMIRT given various security settings on their system (e.g., if ''cookies'' and related authentication information is not stored or otherwise accessible at run-time). The flexMIRT *User's Guide* (Houts & Cai, 2015) provides detailed information about the installation process with recommendations for handling typical operating system or configuration-specific complications.

User Interface and Usability

The graphical user interface (GUI) is very basic with four menu options available in the start-up window: (a) ''File'' for beginning a new flexMIRT project, opening a prior project, or loading a data file for the current analysis project; (b) ''Edit'' for carrying out basic text editing for any of the active windows; (c) ''flexMIRT'' for running the analyses; and (d) ''Help'' which provides access to the User's Guide or to online resources. Additional windows are automatically opened as needed by flexMIRT. The operation of the program is also very straightforward.

Users must create their data file outside of flexMIRT.[1] The scored item response files can be in space-, tab-, and comma-delimited file formats. The software assumes that item scores are ''zero-based'' (i.e., coded as 0,1,2, . . .). Item response scores can be conveniently recoded by one or two syntax commands—including reverse coding of rating-scale data. Note that flexMIRT cannot handle compressed, fixed-column formatted response data strings without spaces (e.g., ''0110011''). However, that is a minor inconvenience as programs like Microsoft Excel® have import data capabilities that employ user-friendly ''wizards'' to convert compressed fixed column response files to distinct variables that can then be saved as space-, tab- or comma-delimited text files. It seems reasonable to suggest that future versions of the software might consider offering a more elaborate data import utility for converting or directly reading other formats.

Similar to many other statistical and psychometric analysis software packages, flexMIRT has its own specialized analysis syntax that is stored in text files. The syntax files can be created within the flexMIRT editor or using any external text editor. The syntax files have four required sections of commands: (a) the ''<**Project**>'' section allows users to enter an analysis title and a description of the analysis; (b) the ''<**Options**>'' section indicates the type of analysis (item calibration, scoring, or simulation) as well as numerical convergence criteria, setting

quadrature points, using multi-core threaded processing on compatible computers, setting output options, and other technical analysis aspects; (c) the ''<**Groups**>'' section specifies the input data file, variable names, the selected variables to be analyzed, and the IRT model(s) of choice—including any multilevel or grouping conditions; and (d) the ''<**Constraints**>'' section contains all relevant parameter constraints (e.g., fixed values or equality constraints) and stipulations for any relevant univariate priors. All four sections must be included, even if default values of the commands and settings for each section are used.

The flexMIRT command syntax is uncomplicated. Commands use a ''*key_word = value(s);*'' format where a terminating semicolon denotes the end of the command. This allows multiline commands. The ''*key_word*'' is a recognized, program-specific command and ''*value(s)*'' may either be program-specific options or user specified values (e.g., ''*Mode = Scoring*;'' or ''*Etol = 0.001*;''). Although the rather verbose flexMIRT syntax is reasonably intuitive, one-, two- or three-character command short-cuts might be a useful feature to incorporate in the future (e.g., ''*mo = sc*'' as an allowable abbreviation for *Mode = Scoring*). In addition, a single-page, printable, quick reference card for the syntax commands and options might be a useful feature to offer online to registered users.

The *flexMIRT User's Manual* (Houts & Cai, 2015) includes a limited number of applied examples that can be modified by end-users for their own analyses (e.g., setting up a three-parameter logistic model calibration). These are supplemented with a growing number of online examples (see ''Program Help'' section). The *Manual* also has a fairly detailed subject index of command statements and other key words that cross-references all of the syntax commands to descriptions or to specific applications in the *User's Manual*. Page numbers appeared to be up-to-date. Chapter 8 of the *Manual* also provides a fairly comprehensive list of syntax commands with technical explanations and references provided for most options.

Program Help

In addition to the *User's Manual*, Vector Psychometrics Group maintains a website with a ''Frequently Asked Questions'' page. Users can also submit questions to flexMIRT@VPGCentral.com. Perhaps a user's forum page might be useful to add in the future.

The VPG website also has an index page titled ''Project Examples.'' This page supplements the examples in the *User's Manual* with syntax examples, including sample data files where relevant.

Capacities

There is no upper limit imposed on the number of items or on the number of respondents. The threaded numerical processing options that flexMIRT supports can also significantly improve the speed of large-scale analyses, if end-users are technically savvy enough to take advantage of those capabilities. Obviously, system-specific memory constraints and ''memory-to-disk'' swaps executed by the operating system can adversely affect program performance for extremely large calibrations or simulation studies.

Ease-of-Use: Running an Analysis

The analysis begins when the ''Run'' (or ''Save and Run'') option is selected from the main ''flexMIRT'' menu option. The progress of the analysis is shown within the main window. Output files are automatically created if the analysis is successfully completed. Note that pre-processing and syntax error checking is fairly minimal for the current software version. As a

result, some syntax errors, analysis misspecifications, or data management issues can ''crash'' the program without extensive feedback provided for trouble-shooting the issue. It is hoped that enhanced error handling will be added to future versions of flexMIRT.

The program can also be run from the Windows command line or from batch files. The latter capability is extremely useful for carrying out multiple routine operational analyses and for simulation research with multiple replications of the analyses.

The analysis outputs are saved as text files. An analysis summary is automatically generated if the program runs to successful termination. If specified in the <**Options**> section, the item parameter estimates and examinee scores are also saved. IRT-based graphics (e.g., plots of item or test characteristic or information functions) are not directly supported as internal flexMIRT capabilities. Rather, outputs from flexMIRT can be called by R program code provided by VPG for generating graphics.

Calibration and Test Scoring Capabilities

FlexMIRT has three analysis modes: (a) calibration, (b) scoring, and (c) simulation. The simulation mode is evaluated separately in the next section of this review. It should be noted that scoring is not unique to that analysis mode. The calibration mode also includes the option to estimate and save examinee scores. By incorporating a separate scoring mode, however, researchers and practitioners can use previously calibrated item parameter estimates to directly compute examinee scores for a different dataset or to produce score tables that allow for convenient number-correct to IRT scale score look-ups (Thissen, Nelson, & Swygert, 2001).

As the program's name implies, flexMIRT was designed to support an extensive array of unidimensional and multidimensional item response theory (UIRT and MIRT) models. Specifically, the program can fit the UIRT one-, two-, three-parameter logistic models (1PLM, 2PLM, and 3PLM) for dichotomous data (e.g., Birnbaum, 1968; Hambleton & Swaminathan, 1985; Lord, 1980; Rasch, 1960; Wright & Stone, 1979). The supported models for multinominal response and polytomous response data include the nominal response category model (NRCM; Thissen, Cai, & Bock, 2010), the rating scale model (RSM; Andrich, 1978), the partial-credit model (PCM; Masters, 1982), the generalized partial-credit model (GPCM; Muraki, 1992), and the graded response model (GRM; Samejima, 1970). flexMIRT's capabilities also extend to multidimensional versions of those models, including the bi-factor model and multilevel item factor models with multiple groups. Common diagnostic classification models (e.g., Rupp, Templin, & Henson, 2010) are also supported in the most recent release of the software.

It does seem relevant to note that flexMIRT uses the factor model parameterization in the exponent for most of its supported IRT models. For example, the well-known 3PL IRT model can be written as follows:

$$P(y_i = 1|\theta) = c_i + (1 - c_i)\{1 + \exp[-Da_i(\theta - b_i)]\}^{-1}, \quad (1)$$

where c_i is the pseudo-guessing item parameter, a_i is the slope parameter, b_i is the item location (difficulty), and D is a scaling constant that allows the logistic response function to closely approximate the cumulative normal probability distribution if D = 1.7. flexMIRT re-parameterizes the 3PL model as

$$P(y_i = 1|\theta) = g + (1 - g_i)\{1 + exp[-a_i\theta - c_i]\}^{-1}, \quad (2)$$

where the pseudo-guessing parameter is re-labeled as $g_i = c_i$ and the intercept is $c_i = -a_i\ b_i$. flexMIRT uses the logit $\lambda_i = \ln[g_i / (1 \text{ ÿ } g_i)]$ in the item parameter input and output files. This re-parameterization and the associated notation in the *flexMIRT User's Manual* (Couts & Cai, 2015) may present a minor complication for some users to adequately understand the content of the various item parameter input and output files.

Before discussing the calibration and scoring modes in flexMIRT, it is important to understand how ''constraints'' are used to better appreciate the inherent versatility built into the program. On one hand, having numerous settings and analysis options (including ''constraints'') may seem a bit daunting for some novice users who are more used to menu-driven psychometric software packages that only support one or two types of IRT model. On the other hand, learning how to specify constraints and set other key program options provides users with enormous array of IRT calibration and scoring possibilities.

Constraints

Parameter constraints play an important role in flexMIRT's modeling elasticity. Virtually any item parameters or population subgroup parameters can be fixed at default or specified quantities or set equal across sets of items, levels, or population subgroups. This opens up many possible capabilities to calibrate and score the data for even hybrid IRT models (e.g., a partially constrained GRM with the slope parameters set equal within ''item families''). Constraints also make it possible to carry out non-equivalent groups linking studies with ''anchor items,'' or to aid in estimating parameters under some of the more complex models using less-than-ideal data—for example, to help resolve estimation convergence issues.

All constraints are obviously specified in the ''<**Constraints**>'' section. The basic syntax includes the type of constraint, the group(s) or variables impacted by the constraint, and the model parameters to be constrained. Probably, the two most common constraint types are (a) fixing or freeing specific parameters and (b) setting two or more parameter estimates to be equal to each other. By default, a fixed parameter is set to zero; however, a corresponding ''*Value*'' statement in the ''<**Constraint**>'' section can assign a non-zero value to the parameter of interest. The ''*Fix*'' statement also provides an easy way to create a simpler model from a more complex one. For example, the popular Rasch model (1PLM) can be implemented as the calibration or scoring model either by constraining the 3PLM slopes (*a* parameters) and pseudo-guessing (*c* or *g*) parameters, respectively, to 1.0 and 0.0, or by using the GRM with the slopes of all items fixed at $a_i = a = 1.0$.

Prior distributions for any of the parameters are also specified in the ''<**Constraints**>'' section. Only the normal, log-normal and two-parameter beta distributions are supported. If not specified, default priors are used where appropriate. The example shown in the ''Software Comparison'' section of this review demonstrates an implementation of priors on the item parameters that allow flexMIRT to closely match the BILOG-MG calibration and scoring results.

Calibration

Calibration requires the ''*Mode* = *Calibration*'' command statement to be specified in the ''<**Options**>'' section of the syntax file. The IRT calibration model(s) of choice are specified at the item level within the ''<**Groups**>'' section. This allows for simultaneously calibrating any combination of models for one or more population subgroups. Obviously, it is not feasible to combine unidimensional and multidimensional models as the latent space for any given calibration must have a fixed number of dimensions. Nonetheless, this capability to employ

different combinations of IRT models allows different item types and mixtures of dichotomous and polytomous data to be jointly calibrated to (a) common scale(s).

Two commands specify the item variables to use in a calibration. ''*Varnames* = [*list of variable names matching input data fields*];'' is required[2] to identify the item and any grouping variables to use in the analysis. The option ''*Select* = [*list of variable names*];'' indicates a particular subset of item variables to analyze. This ''*Select*'' command option makes it possible for other (unused) variables to be excluded without needing to regenerate the data file (e.g., dropping from a calibration any items deleted during an answer key validation or eliminating items that show significant data-model misfit). All of the items specified in by the ''*Varnames*'' command are analyzed if the ''*Select*'' command is omitted.

Specifying the desired model(s) is relatively simple. For most applications, the user creates the ''*Model*([*list of variable names using this model*]) = *model_options;*'' statements within the ''<**Groups**>'' section. A single model statement can be applied for all items or multiple model statements can each be applied to assign specific models to various subsets of items. For example, to calibrate a 50-item test with input variables identified as ''*VarNames = i1-i50*;'' using the 3PLM and GRM, we might need three ''*Model* = '' statements: (a) ''*Model(i1-i45) = ThreePL*;'' (b) ''*Model(i45-i49) = Graded(4)*;'' and (c) ''*Model(i50) = Graded(6).*''

The model specification syntax differs slightly for dichotomous and polytomous models. Additional command statements and constraints may further be needed in the ''<**Groups**>'' section if a multidimensional analysis is carried out. The 3PL, GPCM, GRM, and NRCM can be directly specified by name: ''=*ThreePL*;'' ''=*Graded*(*#_Categories*);'' ''=*GPC*(*#_Categories*);'' or =*Nominal*(*#_Categories*);''. The ''#_Categories'' indicates the number of raw score categories for that corresponding group of item variables. It is important to understand that the 1PL, 2PL RSM, and PCM models are treated as special cases of the NRCM, GRM, or GPCM. That is, these models need to be specified using a more general model and then implemented using appropriate constraints on the parameters (see discussion of ''Constraints'' section).

Scoring

Scoring is very straightforward to perform in flexMIRT. It can be performed simultaneously with or separately from an item calibration. The syntax for scoring is fairly intuitive depending on the type of scoring requested and whether scoring is performed simultaneously or separately from calibration. When the ''*Mode = Scoring*;'' option is specified, a ''*ReadPRMFile = ''Item_Parameter.txt''*;'' is also needed to import the item parameter estimates. The flexMIRT user's guide contains detailed information about properly formatting the item parameter file. (Note: in the calibration mode that same command statement would import the item parameter estimates as either starting values or as fixed values in an ''anchored'' calibration.)

flexMIRT can generate the three most popular types of IRT examinee scores: (a) maximum likelihood estimates (MLEs), (b) *expected a posteriori* (EAP) estimates, and (c) *maximum a posteriori* (MAP). The syntax is very basic (e.g., ''*Score = EAP*;''). In addition, the program can estimate sum score conversion tables (Thissen et al., 2001) and multiple imputations from an estimated posterior distribution (which is only available when the Metropolis–Hasting Robbins–Monro (MH-RM) algorithm is used for calibration). Various ''*Save*???'' command statements are also available in the ''<**Options**>''.

Estimation and Special Analysis Features

The two primary estimators for the item calibrations are (a) marginal maximum likelihood estimation (MMLE) implemented via the expectation–maximization (EM) algorithm (Bock &

Aiken, 1981) and (b) the MH-RM estimation algorithm (Cai, 2010a, 2010b). As noted above, the examinee scoring algorithms include MLE, Bayesian EAP and MAP scores (e.g., Mislevy, 1984, 1986), and multiple imputations (i.e., random draws from a posterior distribution).

The dimension reduction is subtle but is an important flexMIRT feature to mention. It appears to be a special case of adaptive quadrature estimation that allows the posterior densities for higher order models to be computed at a more computationally efficient subset of quadrature points (nodes) to approximate the necessary numerical integration over multiple dimensions. For example, a two-dimensional calibration with 11 quadrature points, K, would have $K^2 = 121$ nodes over which to sum to approximate the double integration. In general, for higher dimensional models, the number of nodes is K^M, where M is the number of dimensions. Therefore, using $K = 11$ quadrature points for $M = 4$ dimensions, the software would need to otherwise carry out the approximate integration during the MMLE EM cycles at 14,641 nodes. In adding to demanding enormous memory resources, continually summing over that large number of nodes could increase the calibration time by an order of magnitude. The dimension reduction feature helps to resolve the well-known ''curse of dimensionality.''[3]

In addition to its extensive multidimensional and multilevel model support, flexMIRT provides multiple methods for estimating item parameter standard errors and an elaborate array of model fit statistical indices for assessing dimensionality, testing model-data fit for individual items, and statistical diagnostic tests of latent variable normality. Specialized options also let users carry out differential item function (DIF) analyses.

Outputs

Most flexMIRT outputs can be generated using the default program settings. Other outputs can be specified in the ''<**Options**>'' section by merely including the output keyword and ''*Save*??? = *Yes*;''. For example, including the command statement ''*SaveINF* = *Yes*;'' automatically produces an output file containing the test information functions at fixed quadrature points. Calibration-specific outputs include various data-model fit indices such as the Akaike information criterion, the Pearson χ^2, the G^2 likelihood ratio test, and the root mean square error approximation (RMSEA). Other goodness-of-fit indices can be requested by adding the command statement ''Gof = Extended;''. The outputs are generally informative, thorough, and intuitive. For users preferring the more traditional 1PLM, 2PLM, or 3PLM parameter estimates, the option ''*NormalMetric3PL* = *Yes*;'' can be added to the ''<**Options**>'' section to generate the popular *a*-, *b*- and *c*-parameters as alternatives to the flexMIRT defaults.

It should be noted that no data field label record (i.e., no variable name header row) is included in the item or examinee parameter estimate output files. The same is true for other types of detailed results outputs printed in column formats. Although the *User's Manual* provides minimally adequate descriptions of each of the output fields, it might be useful to provide the field names header row as an output option.

Graphics and Interfacing flexMIRT With R

flexMIRT does not have built-in graphic plotting functions for item/test characteristic curves and information function plots. However, generic R code is provided to produce some of the more common unidimensional IRT graphs for both dichotomous and polychromous item response models. The downloadable code must be executed from within the R environment. The R scripts read the item parameter estimates from flexMIRT and compute and plot the relevant item or test functions for a particular model. Additional aesthetics can be added to the R

code as needed. Guided information about using the R scripts is also available from the website support page document, *flexMIRTplottingManual* (VPG, 2015).

Simulation Studies in flexMIRT

Unlike most IRT calibration software packages, flexMIRT provides a comprehensive simulation mode that allows researchers to generate response data under any of the models supported by the software, including the multidimensional, bi-factor, and multilevel models. This simulation capability lets researchers carry out model-based research where both the data generation and analyses can be handled in the same software package. A command-line start-up option also makes it possible to execute batch runs with multiple analysis replications using flexMIRT. As discussed below, the simulation function may have some minor short-comings but is nonetheless commendable.

flexMIRT has two rather distinct methods for simulating and analyzing model-generated data: (a) directly running a simulation in the flexMIRT workspace window by submitting one or more simulation mode control files—including the option of using command-line-submitted batch files, and (b) using the flexMIRT simulation ''wizard.'' The latter ''wizard'' is an option under the ''flexMIRT'' menu—what the *User's Manual* calls the ''*Internal Simulation Function*'' (ISF). There appear to be unavoidable trade-offs between these two methods: relative simplicity of setting up and running a potentially large number of simulation replications but with limited output versus spending significantly more time up front to create and run multiple syntax files, but then obtaining a more extensive and detailed set of results that can be analyzed outside of flexMIRT.

The ISF Approach

The ISF can only be executed from within the flexMIRT program interface. It is a relatively simple way to set up and run various straightforward simulations such as IRT parameter recovery studies. As alluded to above, the ISF favors ease-of-use over providing detailed outputs. Users specify a ''data generating model'' and one or more analysis models that are then fit to the generated data (i.e., ''fit model 1,'' ''fit model 2,'' etc.). The data-generation model and the analysis model(s) need not be the same. Different analysis (fit) models can also be specified for comparative analyses. The user interface requires one control file for each model to be included in the analysis as shown in Figure 1. The user must also specify the number of simulation replications (e.g., 20) to run and an output file.

The *flexMIRT User's Manual* does not provide any type of detailed explanations of the required format of the input files. Instead, users must intuit that the inputs in the ''Data Generating Model'' should conform to the ''Mode = Simulation;'' format (see ''Simulation Mode by Syntax Files'' section). The specified ''Fit Model #'' syntax files most often are calibration files (e.g., see ''*Mode = Calibration;*'' in the ''Calibration'' subsection). The *User's Manual* more or less recommends that users engage in some trial-and-error attempts using one replication until they both understand the required inputs and can read/interpret the output files.

A relatively simple simulation study was run using the ISF to generate 20 replications of a GRM simulation with $N = 2{,}000$ for eight items scored 0 to 3 points. The data-generating model parameters are shown in Table 1, where c_{ik} are the category cumulative response function intercepts ($k = 1, \ldots 3$) under the factor-model parameterization of the GRM.

The actual content of the syntax files for the ''Data Generating Model'' and the ''Fit Model 1''—respectively, specified as ''Simulate_GRM8.txt'' and ''Calibrate_GRM.txt'' files in Figure 1—is reproduced in the appendix. Once specified, the user merely needs to click on the

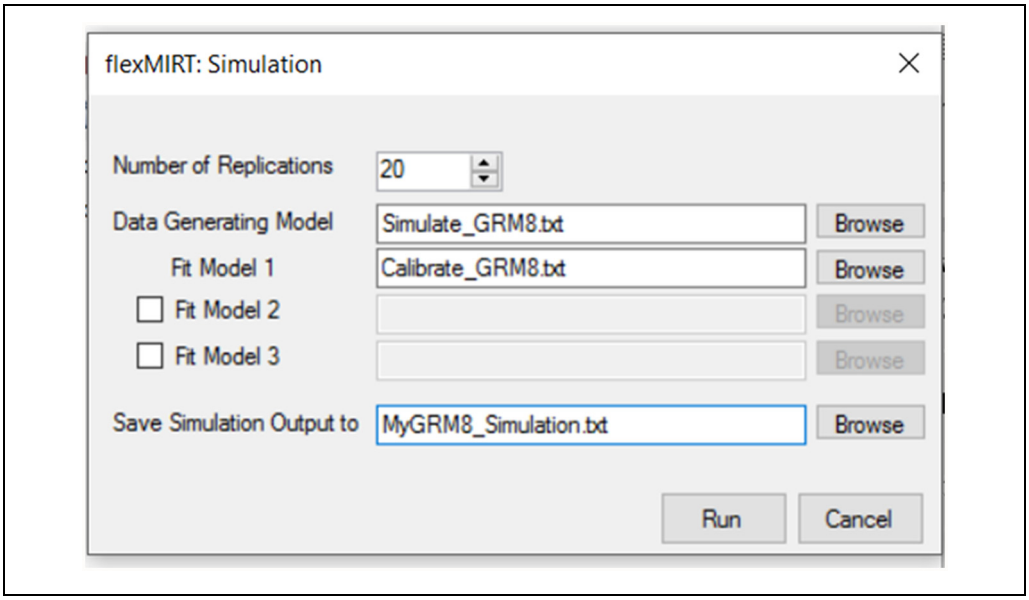

Figure 1. flexMIRT ISF Inputs.
Note. ISF = Internal Simulation Function.

Table 1. Generating GRM Parameters for a Simulation With Eight Items.

Item	Slopes	GRM intercept parameters		
	a	c_1	c_2	c_3
1	0.65	2.00	1.00	0.00
2	0.90	2.00	1.00	0.00
3	1.10	1.50	0.25	−0.50
4	1.25	1.50	0.25	−0.50
5	0.70	1.00	0.00	−1.00
6	1.00	1.00	0.00	−1.00
7	1.20	0.50	−1.00	−2.00
8	1.50	0.50	−1.00	−2.00

Note. GRM = graded response model.

''Run'' button. This 20-replication simulation ran in 5.7 s on a laptop computer with 8 gigabytes of memory and an Intel I7 dual-core processor. Figure 2 summarizes the simulated GRM item parameter estimation errors.

The ISF generates a scored-response data file for each of the replications. The replication sequence number is appended to the ''*File* = *outputfile.name*'' specified under the <**Groups**> section of the simulation data-generating model file (''Simulate_GRM8.txt'' in Figure 1—also see the sample syntax in the appendix). Therefore, 20 data files were generated for this simulation (''My_Sim-0.dat,'' ''My_Sim-1.dat,'' etc.). Each replication file contains (space-delimited) response scores for the items, a record number, and the generated (true) theta value for each simulated examinee.

Other than showing each of the replication analyses on-screen, the ISF provides only limited outputs.[4] To somewhat further confound the end-user—at least not without intense scrutiny of

the output file and comparisons to examples in the *flexMIRT User's Manual*—there is NO field (variable) header row in the file, nor are the contents of each data column labeled in any of the outputs. The *Manual* rationalizes the lack of a variable label header row as facilitating subsequent analyses in software such as R. However, it would be extremely useful to offer a column labeling option or to at least automatically generate a list of variables and their column positions in another output file.

To better understand the nature of this criticism, consider the file content of the output file for the present simulation example (i.e., the content of the ''MyGRM8_Simulation.txt'' output file in Figure 1). The first and last six columns of that output file contained some fixed-position outputs—most of which are data-model statistics such as log-likelihood G^2 and χ^2—and 66 additional variable-position columns: 32 GRM item parameter estimates in intercept-first order, the mean and standard deviation of the generated latent distribution for that replication, and, finally, the 32 standard errors for each of the item parameter estimates. It was definitely not intuitive to figure out which values were in which columns. As suggested above, this could be an area for some serious improvement.

It also seems fair to point out that the ISF has no apparent or easy way to save the estimated θ scores from each of the replications for subsequent analysis outside of flexMIRT. At best, users are provided with various item and group-level aggregate fit statistics in the ISF output file. Of course, one could create and specify in a unique calibration control file for each of the replication datasets ''*SaveSCO* = *Yes*;'' and ''*SavePRM* = *Yes*;'' in the <Options> section. But creating and running those files would seem to offset the ease-of-use benefits of the ISF.

Simulation Mode by Syntax Files

Simulation syntax files differ from calibration and scoring files by indicating ''*Mode* = *Simulation*;'' in the <**Options**> section. An in-depth description of the simulation options is beyond the scope of this review. Suffice to say, the item parameters can either be generated by flexMIRT in the <**Constraints**> section (see sample syntax in the appendix for the Simulate_GRM8.txt file used for the ISF example) or input from an external file.

The *flexMIRT User's Manual* and the VPG website[5] have a reasonable number of examples of simulation syntax files. More extensive output is created when the syntax control files are used. In addition, the estimated item parameters, scores, and other outputs can be requested by including appropriate ''*Save???* = *Yes*;'' statements in the <Options> section of each syntax file.

Software Comparison

A comprehensive set of comparisons is well beyond the scope of this review. Nonetheless, a relatively straightforward 3PL comparison was carried between flexMIRT and BILOG-MG (Zimowski, Muraki, Mislevy, & Bock, 2003) to demonstrate some of the very minor differences between these packages. BILOG-MG has a well-deserved reputation as the de facto standard for IRT 2PL and 3PL calibrations. It therefore made sense to include it in this study.

Data were generated based on the 3PLM in a third-party software package, GenData3PL (Luecht, 2009). The simulation included 50 items with *a*-parameters sampled from a log-normal distribution with $\mu[\ln(a)] = -0.1$ and $\sigma[\ln(b)] = 0.2$; *b* parameters sampled from a normal distribution with $\mu(b) = 0.0$ and $\sigma(b) = 1.5$; and *c* parameters drawn from a beta distribution with $\alpha = 4$ and $\beta = 20$. Two thousand θ scores were randomly sampled from a unit-normal distribution, $\theta \sim NID(0,1)$.

Table 2 summarizes the generating parameters for the 50 items and for the 2,000 sampled θ values. The number-correct raw scores for the generated item response scores are summarized in the rightmost column. The scale reliability (Cronbach's α) of the raw scores was .877.

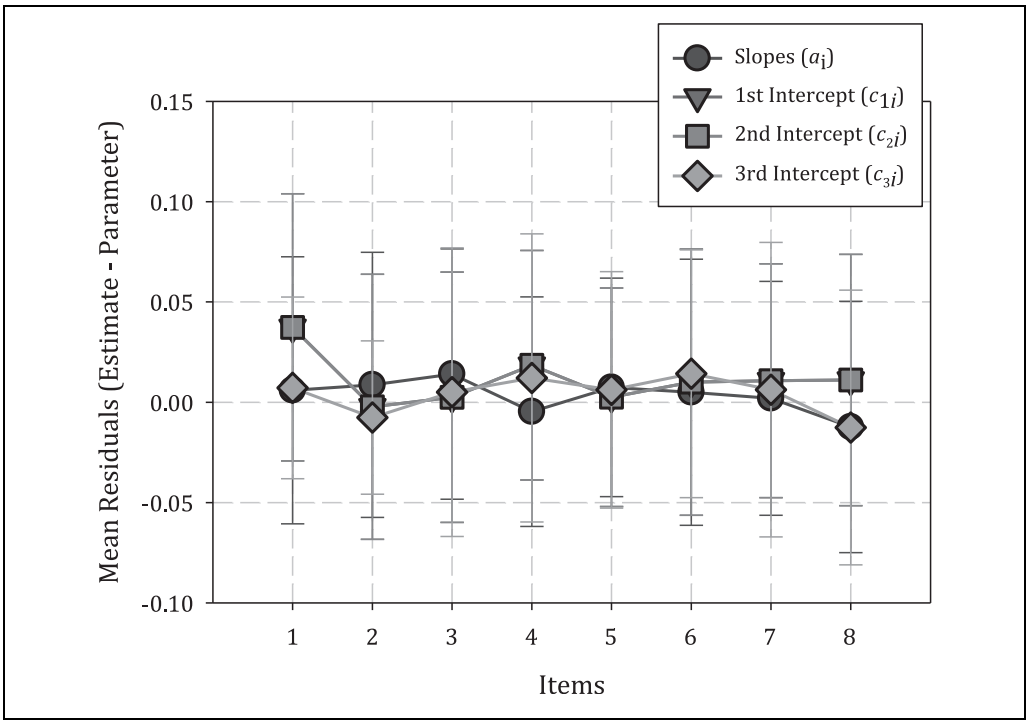

Figure 2. Means (symbols) and standard deviations (error bars) of the GRM item parameter estimation errors for eight items.
Note. GRM = graded response model.

Table 2. Descriptive Statistics for the Simulation Study.

Statistics	*a*	*b*	*c*	θ	*x*
Count	50			1,000	
M	0.897	0.000	0.151	0.063	29.19
SD	0.179	1.592	0.053	0.999	7.92
Minimum	0.521	−3.360	0.044	−3.000	5
Maximum	1.318	3.647	0.307	2.994	49

The default BILOG-MG priors were used (DeMars & Juris, 2012; Zimowski et al., 2003). Figure 3 shows the relevant flexMIRT syntax. The ''*Prior*'' statements in the <**Constraints**> section are especially relevant because they exactly match the BILOG-MG default priors for the slope and pseudo-guessing parameters. The ''*NormalMetric3PL = Yes;*'' statement is also important for this example because it forces flexMIRT to generate the more traditional 3PLM item parameter estimates (see Equation 1).

Figure 4 shows the comparative item parameter estimates, *a, b*, and *c*, for BILOG-MG (horizontal axes) and flexMIRT (vertical axes). Although there are minor differences between the calibrated *a*- and *c*-parameter estimates that no doubt stem from differences in the number of quadrature points and various nuanced differences in the implementation of the marginal maximum likelihood solutions and E-M algorithm, the results are compellingly similar. The

```
<Options>
  Mode = Calibration;
  Etol = 0.005;
  NQuadrature=41,5.5;
  SavePRM=Yes;
  SaveSCO=Yes;
  Score=EAP;
  NormalMetric3PL=Yes;
  Quadrature = 49, 6.0;
  MaxE = 1000;
  MaxM = 250;

<Groups>
  %Group%
  File = "3PL_50Items_Data.csv";
  Varnames = PersonID,  i1 - i50;
  Select = i1-i50;
  N = 2000;
  Ncats(i1-i50)  = 2;
  Model(i1-i50) = ThreePL;

<Constraints>
  Prior (i1-i50),  Guessing :  Beta (5.0,17.0);
  Prior (i1-i50), Slope : LogNormal(0.0,0.5);
```

Figure 3. Partial flexMIRT syntax to mimic BILOG-MG calibration.
Note. EAP = expected a posteriori.

corresponding product–moment correlations are $r_{a,a'}$ = .909, $r_{b,b'}$ = .998, and $r_{c,c'}$ = .977. Of course, even given that magnitude of similarity, we would probably not go so far as to claim that the two calibrations produce isomorphic results.

The EAP estimates of θ correlated at 1.0 to three decimal places of precision. Nonetheless, Figure 5 shows that there were still some very minor differences in the score estimates. The symbols show the mean residual (difference between the flexMIRT and BILOG-MG EAPs) with the error bands set to the standard deviation of the residuals. The plots show the differences as conditional distributions of the residuals at every total raw-score point on the simulated test.

Discussion

This review attempted to provide a fair evaluation of flexMIRT from various perspectives that are hopefully relevant to academic researchers and practitioners, alike. In general, the software lives up to its name by providing exceptional versatility in fitting a variety of unidimensional and multidimensional models to dichotomous and polytomous data. The syntax allows many different features and program options to be mixed and matched to create an enormous number of possible analysis configurations.

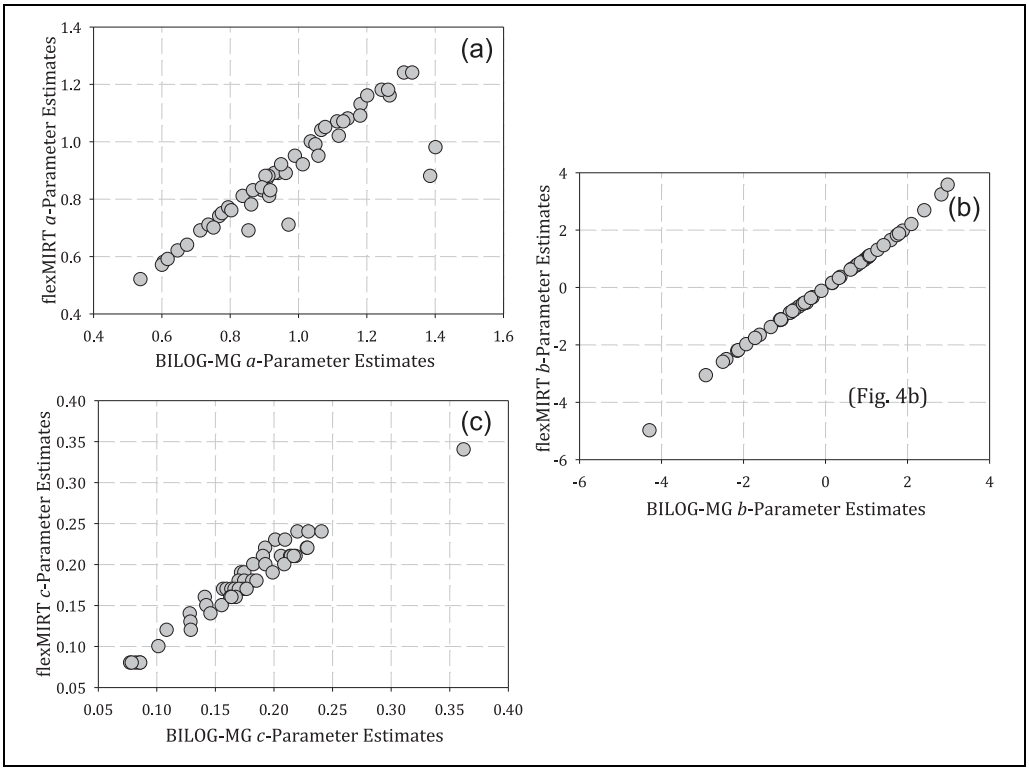

Figure 4. Bivariate scatter plots of flexMIRT by BILOG-MG (a) *a*-parameter estimates, (b) *b*-parameter estimates, and (c) *c*-parameter estimates.
Note. EAP = expected a posteriori.

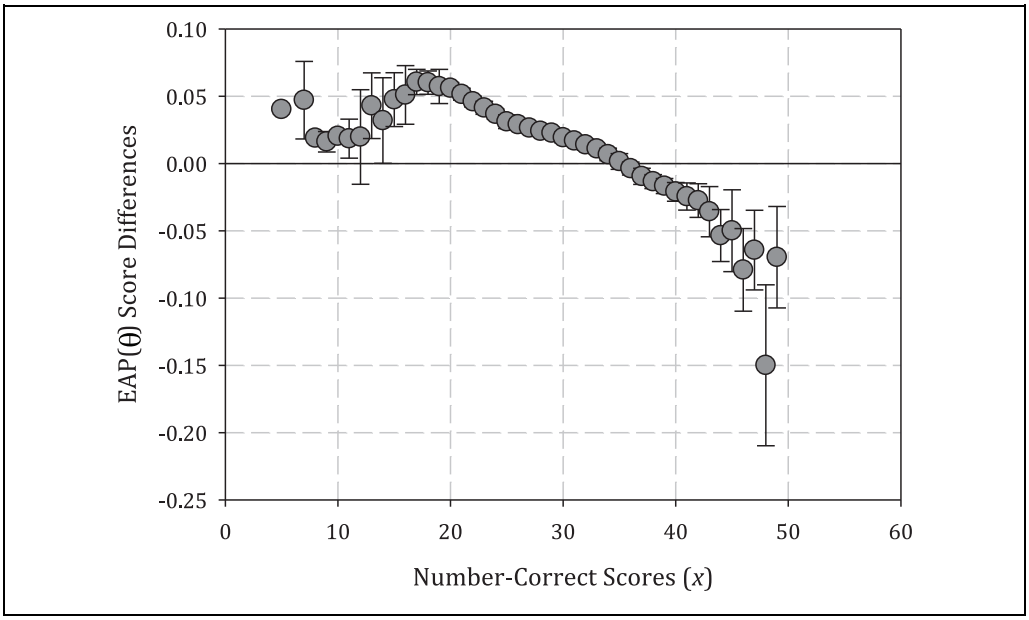

Figure 5. Error bands depicting the distributional differences in flexMIRT and BILOG-MG EAP estimates conditional on number-correct score.
Note. EAP = expected a posteriori.

The simulation capabilities built into flexMIRT are also impressive. Although there may be room for improvement relative to some of the (lack of) input descriptions in the *User's Manual* and some issues discussed regarding labeling of simulation output results, the package performed quite well, once its syntax and other nuances were reasonably well understood. The online resources, especially the sample syntax and data files, were particularly helpful to supplement information in the *User's Manual*.

flexMIRT already is a useful research tool. However, it promises to also be a serious contender for many operational applications ranging from more traditional 1PLM, 2PLM, 3PLM calibration and linking studies to mixed model analysis using combinations of dichotomously and polytomously scored items. flexMIRT should also prove useful as the psychometric community transitions from basic research on multidimensional and multilevel IRT modeling to more operational applications.

Appendix

Simulate_GRM8.txt (Simulation Syntax)

```
<Project>
       Title = ''Simulate Data for 8 GRM(4 Category Items)'';
       Description = ''8 GRM(4) Items; N=2000 ~ (0,1)'';
<Options>
       Mode = Simulation;
       Rndseed = 2671;
<Groups>
       %Group1%
       File = ''MySim.dat'';
       Varnames = i1-i8;
       N = 2000;
       Model(i1-i8) = Graded(4);
<Constraints>
       Value (i1),Slope,0.65;
       Value (i2),Slope,0.90;
       Value (i3),Slope,1.10;
       Value (i4),Slope,1.25;
       Value (i5),Slope,0.70;
       Value (i6),Slope,1.00;
       Value (i7),Slope,1.20;
       Value (i8),Slope,1.50;
       Value (i1,i2),Intercept(1),2.0;
       Value (i1,i2),Intercept(2),1.0;
       Value (i1,i2),Intercept(3),0.0;
       Value (i3,i4),Intercept(1),1.5;
       Value (i3,i4),Intercept(2),0.25;
       Value (i3,i4),Intercept(3),-0.50;
       Value (i5,i6),Intercept(1),1.0;
       Value (i5,i6),Intercept(2),0.0;
       Value (i5,i6),Intercept(3),-1.0;
       Value (i7,i8),Intercept(1),0.5;
       Value (i7,i8),Intercept(2),-1.0;
       Value (i7,i8),Intercept(3),-2.0;
```

Calibrate_GRM8.txt (Calibration Syntax)

```
<Project>
        Title = ''Calibrate Simulated GRM Data Scored 0 to 3'';
        Description = ''8 Items, N=2000 Syntax Only'';
<Options>
        Mode = Calibration;
        SaveScore=Yes;
        SavePRM=Yes;
        Etol=0.001;
<Groups>
        %Group1%
        File=''MySim.dat'';
        Varnames = i1-i8, personID, theta;
        Select = i1-i8;
        N = 2000;
        Ncats(i1-i8) = 4;
        Model(i1-i8) = Graded(4);
<Constraints>
```

Authors' Note

This software review was originally conducted as part of a graduate measurement course project in the Educational Research Methodology program at the University of North Carolina at Greensboro.

Acknowledgments

The authors thank Richard M. Luecht for providing guidance in the development of the review. The authors also thank R. J. Wirth and Li Cai for making the flexMIRT software available for the course and for agreeing to this review.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Notes

1. The exception is simulations. flexMIRT generates data file for model-based simulations.
2. If a ''*Header* = *True*'' command were available, the required ''Varnames'' statement could be made optional.
3. This phrase is attributed to D. Thissen (personal communication, circa, 2002).
4. If the calibration (''Fit Model #'') file has ''SaveSCO = Yes;'' and/or ''SavePRM = Yes;'' options specified, flexMIRT will only create the estimated scores and/or estimated item parameters files for the final replication. It appears that these files are merely overwritten at each iteration. Perhaps Vector Psychometric Group will add the same sequencing option used for producing the multiple data files to the calibration and/or scoring outputs as well.
5. See ''Syntax Examples'' at www.vpgcentral.com/software/irt-software/support/

References

Andrich, D. (1978). A rating formulation for ordered response categories. *Psychometrika*, *43*, 561-573.

Birnbaum, A. (1968). Some latent trait models and their use in inferring an examinee's ability. In F. Lord & M. Novick (Eds.), *Statistical theories of mental test scores* (pp. 395-479). Reading, MA: Addison-Wesley.

Bock, R. D., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, *46*(4), 443-459.

Cai, L. (2010a). High-dimensional exploratory item factor analysis by a Metropolis-Hastings Robbins-Monro algorithm. *Psychometrika*, *75*, 33-57.

Cai, L. (2010b). Metropolis-Hastings Robbins-Monro algorithm for confirmatory item factor analysis. *Journal of Educational and Behavioral Statistics*, *35*, 307-335.

Cai, L. (2017). flexMIRT®: Flexible multilevel multidimensional item analysis and test scoring (version 3.51) [Computer software]. Chapel Hill, NC: Vector Psychometric Group.

DeMars, C., & Juris, D. (2012). Software note: Using BILOG for fixed-anchor item calibration. *Applied Psychological Measurement*, *36*, 232-236.

Hambleton, R. K., & Swaminathan, H. (1985). *Item response theory: Principles & applications*. Boston, MA: Kluwer.

Houts, C. R., & Cai, L. R. (2015). *flexMIRT user's manual version 3.5: Flexible multilevel multidimensional item analysis and test scoring*. Chapel Hill, NC: Vector Psychometric Group.

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

Luecht, R. M. (2009). Gen3PLData [Computer software]. Greensboro: University of North Carolina at Greensboro.

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, *47*, 149-174.

Mislevy, R. (1984). Estimating latent distributions. *Psychometrika*, *49*, 359-381.

Mislevy, R. (1986). Bayes modal estimation in item response models. *Psychometrika*, *51*, 177-195.

Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. *Applied Psychological Measurement*, *16*, 159-176.

Rasch, G. (1960). Probabilistic models for some intelligence and achievement tests. *Copenhagen: Danish Institute for Educational Research*.

Rupp, A. A., Templin, J., & Henson, R. A. (2010). *Diagnostic measurement: Theory, methods, and applications*. New York, NY: The Guilford Press.

Samejima, F. (1970). Erratum estimation of latent ability using a response pattern of graded scores [Monograph No. 17]. *Psychometrika*, *35*, 139.

Thissen, D., Cai, L., & Bock, R. D. (2010). The nominal categories item response model. In M. Nering & R. Ostini (Eds.), *Handbook of polytomous item response theory models: Developments and applications* (pp. 43-75). New York, NY: Taylor & Francis.

Thissen, D., Nelson, L., & Swygert, K. A. (2001). Item response theory applied to combinations of multiple-choice and constructed-response items and approximation methods for scale scores. In D. Thissen & H. Wainer (Eds.), *Test scoring* (pp. 293-341). Hillsdale, NJ: Lawrence Erlbaum.

Vector Psychometric Group. (2015). *flexMIRT plotting manual*. Retrieved from www.vpgcentral.com/wp-content/uploads/2014/03/flexMIRTplottingManual.pdf

Wright, B. D.; & Stone, M. H. (1979). *Best test design: Rasch measurement*. Chicago, IL: MESA Press.

Zimowski, M. F., Muraki, E., Mislevy, R. J., & Bock, R. D. (2003). BILOG-MG 3 for Windows: Multiple-group IRT analysis and test maintenance for binary items [Computer software]. Skokie, IL: Scientific Software International.