A Bayesian Student Model without Hidden Nodes and Its Comparison with Item Response Theory

Michel C. Desmarais and Xiaoming Pu, *École Polytechnique de Montréal*
michel.desmarais@polymtl.ca, xiaoming.pu@polymtl.ca

Abstract. The Bayesian framework offers a number of techniques for inferring an individual's knowledge state from evidence of mastery of concepts or skills. A typical application where such a technique can be useful is Computer Adaptive Testing (CAT). A Bayesian modeling scheme, POKS, is proposed and compared to the traditional Item Response Theory (IRT), which has been the prevalent CAT approach for the last three decades. POKS is based on the theory of knowledge spaces and constructs item-to-item graph structures without hidden nodes. It aims to offer an effective knowledge assessment method with an efficient algorithm for learning the graph structure from data. We review the different Bayesian approaches to modeling student ability assessment and discuss how POKS relates to them. The performance of POKS is compared to the IRT two parameter logistic model. Experimental results over a 34 item UNIX test and a 160 item French language test show that both approaches can classify examinees as *master* or *non-master* effectively and efficiently, with relatively comparable performance. However, more significant differences are found in favor of POKS for a second task that consists in predicting individual question item outcome. Implications of these results for adaptive testing and student modeling are discussed, as well as the limitations and advantages of POKS, namely the issue of integrating concepts into its structure.

Keywords. Bayesian inference, adaptive testing, student models, CAT, IRT, POKS

INTRODUCTION

Computer Adaptive Testing (CAT) applications are probably the earliest examples of the use of intelligent user modeling techniques and adaptive interfaces in educational applications. The principle behind CAT is to adjust the test items presented to the user's knowledge, or, using CAT terminology, to adjust the item characteristics to the examinee's ability level. Akin to the architectures of adaptive systems, CAT systems analyze the behaviour of the user to build a dynamic model of his/her knowledge state and choose the next item that is most appropriate for this state. In the specific context of CAT, the most appropriate items are the ones that will allow the system to determine, with the least number of test items administered, if the examinee is a "master" or a "non-master" with respect to the measured ability. In the context of adaptive interfaces in general, it could imply, for example, adapting a tutor's didactic content and strategy, adapting hyperlinks of a documentation system, or even adapting some query retrieval results.

The original theory behind CAT is the Item Response Theory (IRT), a framework introduced by Birnbaum (1968) and by Lord and Novick (1968). This theory was later refined by a number of other researchers since its introduction (see van der Linden & Hambleton, 1997). More recently, the Bayesian modeling approach has also been applied to model an examinee's ability based on test item responses. This interest in Bayesian modeling has come not only from researchers in educational testing, such as Rudner (2002) and Mislevy and Gitomer (1995), but also from researchers in adaptive interfaces and user modeling (see, for example, Conati, Gertner, & VanLehn, 2002). It has now become one of the major probabilistic user modeling techniques. This paper focuses on the link between these two fields, namely IRT and the Bayesian graph models student modeling techniques. We compare each approach and conduct a comparative performance evaluation between IRT and one such Bayesian modeling approach named POKS (Partial Order Knowledge Structures).

POKS (Desmarais, Maluf, & Liu, 1995) is particularly well suited for this comparison because, akin to the IRT approach, it does not necessarily require a knowledge engineering effort to build the model but, instead, relies on statistical techniques to build and calibrate the model. Indeed, by relying solely on observable nodes to build a graph model of item-to-item relations, there is no need to define latent skills behind each test item. The process then becomes very similar to IRT for which no knowledge engineering effort is required as a single latent skill (hidden node) is assumed for every test item.

By building links between items themselves, POKS differs from most Bayesian graph models and Bayesian networks developed for student modeling, which rely on hidden nodes such as concepts or abilities. However, in order to provide the detailed diagnostic capabilities required by most intelligent learning environments, concepts and skills must be included within POKS. Exploration of POKS structures that include concepts is provided in Desmarais, Meshkinfam, and Gagnon (2005). The introduction of concept nodes and abilities in POKS and how it compares with other Bayesian models or multidimensional-IRT (MIRT) approaches will be revisited in the section entitled *Automated learning constraint*. For the purpose of comparing POKS with IRT, we limit POKS structures to item-to-item nodes only. Doing so puts both methods on the same footing, namely that of being entirely driven from empirical data. It also allows the proper evaluation of the strength of item-to-item structures for knowledge assessment.

The next two sections provide a basic overview of the IRT and Bayesian modeling approaches. It is followed by a more detailed description of the specific techniques involved, IRT and POKS.

COMPUTER ADAPTIVE TESTING AND IRT

The prevalent means of conducting CAT is based on the Item Response Theory (IRT). In IRT, the probability that an examinee correctly answers a given test item is modeled by an 'S' shaped curve such as the one in Figure 1. This curve is named Item Characteristic Curve (ICC) and it is meant to represent the probability of an item as a function of the student's ability level. The more skilled the individual is, the more likely the item will be mastered by this person. The shape of the probability distribution is modeled by two parameters: the item's *difficulty* level and the item's *discrimination* factor. This is the so

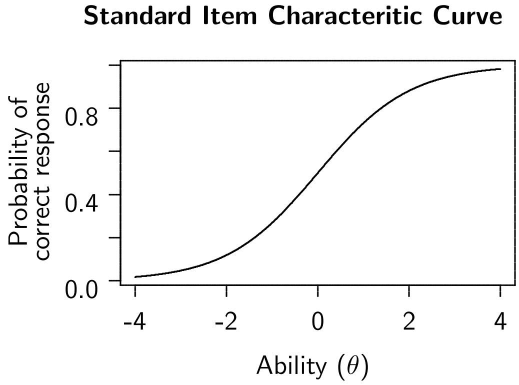

Fig.1. Item characteristic curve ($a = 1, b = 0$).

called "two-parameter model"[1]. These two parameters, *difficulty* and *discrimination*, can be estimated from data of each test item. Typically, parameters are estimated by a maximum likelihood approach or by a least square fit (Baker, 1992). Figure 1 illustrates the typical 'S' curve corresponding to an item of difficulty $b = 0$ (average difficulty) and discrimination $a = 1$.

Once the parameters of the ICC curve are determined for all test items, it becomes possible to derive the examinee's most likely ability level given a set of item responses. Typically, the ability is estimated with a maximum likelihood model (see the section on *IRT-2PL model*); however, a number of methods have been proposed and investigated for this task (Baker, 1992).

The CAT process loop

The ability level is updated after each response obtained from the examinee during the CAT process loop:

1. Estimate the most likely ability level given by the previous responses (if no previous response, take the sample's average score as the initial examinee's ability estimate).

2. Stop if the probability that the examinee is a master falls above or below given boundaries, otherwise continue to step 3.

3. Find the most informative item given the estimated ability level.

4. Present and evaluate the answer to this item. Return to step 1.

In addition to step 1 (ability estimation) the current study investigates the influence of different strategies in step 3, namely the means to identify the most informative item. Two means of choosing the most informative item are investigated for the Bayesian modeling approach: the *information gain* approach and the Fisher information measure. However, we do not focus on step 2, the criteria for

[1] A three parameter model is also defined and it includes an additional chance factor parameter, whereas the "one parameter model" omits the discrimination factor.

deciding on mastery or not. Instead, we simply look at the accuracy of the ability estimation as a function of the number of items posed. This allows us to determine, after every number of items presented, the ratio of correctly classified examinees, for example.

Item selection

There are numerous measures for finding the most informative questions, such as (1) the *item information function* (Birnbaum, 1968), also known as the Fisher information, (2) the *minimum information cost* (Lewis & Sheehan, 1990; Vos, 1999), (3) the *information gain*, or, finally, (4) the relative entropy, also known as the Kullback-Leibler distance. The reader is referred to Rudner (2002) for a comparative study of some of these measures. We describe later the two measures used in this study, namely the *Fisher information* and the *information gain*.

Effective adaptation of the items presented allows the system to determine the examinee's ability level with the least number of items, or, in accordance to the objective of CAT, to make a decision on whether the test subject has achieved mastery or not.

Note that choosing the most informative item is only one of many alternative strategies. The choice could also be determined using other considerations, such as the need to randomize and diversify the items presented across examinees, or to adapt item difficulty to ability level. Moreover, the choice of the items administered could be outside the control of the system. For example, the system could be in a non-intrusive, observational mode, as it is often the case in advice giving interfaces. Regardless of the means by which the item responses are chosen or collected, the underlying IRT ability assessment model can always be used to adapt some other aspect of the user interface, akin to the purpose of any user modeling module and such as those used for adaptive hypertext and courseware (Brusilovsky, Eklund, & Schwarz, 1998). However, in the context of the current study, we will follow the usual CAT goal of assessing ability with the least number of questions.

BAYESIAN MODELING APPROACHES TO CAT AND STUDENT MODELING

We will return to the IRT approach in the next section to provide the details on the specific algorithms used in this study. Let us now turn to the Bayesian approach to student modeling and describe how this approach is applied to CAT.

Bayesian modeling and Bayesian networks

Bayesian modeling provides a mathematical framework by which we can compute the probability of a certain event given the occurrence of a set of one or more events. For example, in CAT, one could compute the conditional probability of mastery of a test item given the previous responses by using samples where such a response pattern was found. This simple but impractical approach relies on the full joint conditional probability table. The problem with this straightforward approach is, obviously, that the number of conditional probabilities grows exponentially with the number of items. The approach quickly becomes impractical because of limited data. For example, computing the probability of correctly answering a specific test item given the answers to the last 10 items would entail constructing a conditional probability table of 2^{10} entries, if we assume each item can take two values, `{success, failure}`. A reliable empirical estimate of this set of conditional probabilities would require thousands of data cases, whereas a subjective estimate is deemed too tedious and unreliable.

There exists a number of means to avoid relying on the full joint conditional probability distribution to perform Bayesian inference. These means will vary according to their assumptions, or according to the constraints they impose on the conditional probabilities in a model.

The Bayesian graph models, and in particular the Bayesian Networks (BN) framework, are amongst the most prevalent approaches to Bayesian modeling. They allows the modeling of only the relevant conditional probabilities and they can rest on a sound mathematical scheme to update the probabilities upon the occurrence of an event in the network (see Heckerman, 1995). Furthermore, the identification of the relevant subset of conditional probabilities, the topology of the network itself, can be derived from empirical data (Heckerman, 1995; Cheng et al., 2002; Liu & Desmarais, 1997).

To reduce the number of conditional probabilities to only the relevant ones while maintaining consistent probability updates from new evidence, BN structures define clear semantics of conditional probabilities and independence relations between nodes in the network. It states that the probability of a node X_i, given the evidence from the nodes' parents $pa(X_i)$, is independent of all nodes except its descendants. Assuming that the vector $X_1, ..., X_i$ represents a specific combination of responses to test items and concept mastery for a given individual, it follows from the above definition of a BN that the probability of this vector is:

$$P(X_1, \ldots, X_k) = \prod_{i=1}^{k} P(X_i | pa(X_i)) \tag{1}$$

where $pa(X_i)$ represents the set of *parent nodes* of X_i in the BN.

For CAT and student modeling, the application of BN and graph models generally consists in modeling the conditional probabilities as a hierarchy of concepts with items as leaf nodes. Figure 2 illustrates a very simple graph structure that, in fact, represents an IRT model. It contains a unique concept node, θ, and a set of item nodes, $X_1, ..., X_n$. The semantics of this networks would state, for example, that the probability of node X_1 is independent of the probability of node X_2 given the ability θ. This definition translates into:

$$P(X_1 | \theta, X2) = P(X_1 | \theta) P(X_2) \tag{2}$$

However, the probability that skill θ is mastered depends on the responses to all item nodes. We return with more details on the IRT model in the section detailing the *IRT-2PL model*.

One of the major advantages of graph models over IRT is that the assessment of the probability of mastery to a test item does not rely on a single trait, namely the examinee's ability level. High level concepts embedded in a graph model constitute a powerful mean of representing a variety of ability dimensions. For example, Figure 2 can be augmented by defining multiple θ over a set of test items, which, in turn, can be organized as a hierarchy or as a directed graph structure with high level θ representing global skills. Moreover, misconceptions can also be included in the structure.

The flexibility and representational power of graph models and their derivatives have been recognized and applied to student modeling by a number of researchers in the last decade (Reye, 2004; Conati

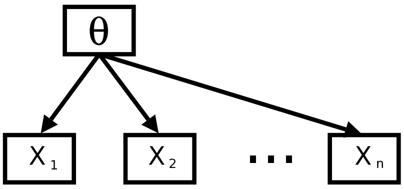

Fig.2. BN structure of an IRT model, where θ is the examinee's ability and $\{X_1, X_2, ..., X_n\}$ are the test items.

et al., 2002; Jameson, 1995; Millán et al., 2000; Mislevy & Gitomer, 1995; Desmarais et al., 1995; Martin & Vanlehn, 1995; Zapata-Rivera & Greer, 2004). They have also been applied more specifically to CAT systems (Vomlel, 2004; Millán & Pérez-de-la-Cruz, 2002; Collins, Greer, & Huang, 1996; VanLehn & Martin, 1997). We will review some of this work in the remainder of this section. The specific Bayesian modeling approach used in the current study will be further described below.

Student Graph Models

Vanlehn, Martin, Conati and a number of collaborators were amongst the most early and active users of BN for student assessment (Martin & Vanlehn, 1995). In the latest of a series of tutors embedding a BN, the Andes tutor (VanLehn, Lynch, Schulze, Shapiro, Shelby, Taylor, Treacy, Weinstein, & Wintersgill, 2005; Conati et al., 2002; VanLehn & Niu, 2001) incorporates a BN composed of a number of different types of nodes (rules, context-rules, fact, and goal nodes). Each node can take a value of "mastered" or "non-mastered" with a given probability. Probabilities can be computed from Bayes posterior probability rule, or in a deterministic binary form (e.g. $P(X=1) \Rightarrow P(Y=1)$), or according to what is known as *leaky-or* and *noisy-and* relations (see Neapolitan, 1998). Most conditional probabilities in the network are subjectively assessed.

In Hydrive, Mislevy and Gitomer (1995) used a BN for assessing a student's competence at troubleshooting an aircraft hydraulics system. The BN is knowledge engineered. Careful modeling of the domain knowledge results is a hierarchy of abilities. Nodes can take multiple values such as `{weak, strong}` or `{expert,good,ok,weak}`. Conditional probabilities are first posited by expert judgment and further refined with empirical data from 40 subjects.

The work of Collins et al. (1996) is amongst the first to create a CAT with a Bayesian network. They use the notion of *granularity hierarchies* to define the BN. Granularity hierarchies are essentially aggregations of concepts or skills into a hierarchy, akin to Mislevy and Gitomer (1995), where the leaves are test items and the root represents the whole subject domain. The BN tested are knowledge engineered from expert knowledge and different topologies are compared. Since the system is a CAT, the choice of the next item is adapted to optimize ability assessment. It is based on a utility measure that yields the item with highest discrimination factor, that is, the item whose difference in ability estimate, $P(\theta|X_i)$, is the highest between a correct and an incorrect answer.

In his unpublished Master thesis, Collins (1996) compares a BN model with an IRT model. He calibrates a BN model with conditional probabilities obtained from an extensive pool of 6000 data cases for a test of 440 items, which in fact corresponds to 44 items replicated 10 times to simulate a large test[2]. Comparison of the BN approach with an IRT model revealed that, after approximately 20 items, the BN approach is more effective in classifying examinees as *master* or *non-master* than the two IRT-based algorithms they compared it with, namely EXPSRT and EXPSRT-2 (Collins, 1996). However, it is not clear what impact the replication of the original 44 items can have on these results and how much this could favor one approach over the other. For example, the non-adaptive paper and pencil test proved more accurate than the IRT and BN approaches, which is unexpected and could be explained by this replication[3].

In a more recent CAT system, Millán and Pérez-de-la-Cruz (2002) defined a hierarchical BN with three layers: concepts, topics, and subjects. A fourth layer links test items to concepts. They used different means of computing the updated probabilities according to the layer. The concepts, topics, and subjects layers use a summative formula to yield an updated probability. New probabilities are a function of weights assigned to evidence nodes according to their importance, which can be a factor of time devoted to a certain topic in a course, for example. At the items level, the probabilities that a concept is mastered is a function of test items. That probability is computed from a conditional probability with parameters modeled from an ICC function such as the one in Figure 1. They tested the accuracy of their approach with simulated students and a test of 60 questions. They found a relatively good performance for assessing mastery of each of 17 different concepts with error rates varying from 3% to 10%.

Learning Graph Models from Data

In contrast with most Bayesian student model approaches, Vomlel (2004) has conducted experiments with empirically derived BN. This work is, to our knowledge, the only experiment using empirical data to construct BN, although it does involve some knowledge engineering effort for categorizing concepts and test items into a hierarchy. Vomlel used HUGIN's PC algorithm (Jensen, Kjærul, Lang, & Madsen, 2002) to calibrate a number of network topologies from 149 data cases of a 20 questions arithmetic tests administered to high school students. The basic topology of the network was constrained based on a knowledge engineering of the domain with experts, but HUGIN was used to refine or define parts of the BN's structure. The BN was composed of a few skills and student misconceptions. Some of the different BN structures tested incorporated hidden nodes that were created by HUGIN's BN induction algorithm. Conditional probabilities were all calibrated from empirical data. The results show that an adaptive test with such BN can correctly identify the skills with an accuracy of approximately 90% after the 9th question and performs significantly better than a fixed question order test.

[2] Note that the BN only contained the 44 original nodes, not 440.

[3] The POKS approach used in the current study would be highly influenced by the replication of items. Replicated items would be aggregated into fully connected nodes, in effect merging them into the equivalent of a single node.

MODELING CONSIDERATIONS FOR THE COMPARISON OF IRT VS. BAYESIAN APPROACHES

When comparing IRT with Bayesian modeling, the question of how the model is built and calibrated (or learned) is a crucial one, as the two approaches differ significantly on that issue. IRT modeling is entirely based on calibration from data and has limited modeling flexibility, whereas Bayesian modeling offers much more flexibility but it involves knowledge engineering efforts that can also be limiting for many applications. These issues are central to the practical utility of student modeling and we discuss them in more details in this section.

Automated Approach Considerations

The IRT models are empirically derived from test data and student expertise is solely defined by a set of observable test items, which usually take on two possible values: mastered or non-mastered[4]. IRT does not rely on any subjective assessment, nor on the ability of a domain expert knowledge engineer, as it requires no human intervention to build the model. The same can be said about POKS with item only node structures.

Such algorithmic techniques, for which the model is learned or induced from empirical data, have important advantages that stem from their amenability to complete automation:

- It avoids the so called "domain expert bottleneck" and is thus more scalable.
- It is not subject to human biases and expert ability to build domain models.
- It lends itself to automated updates when new items are added to a test (a very common situation for qualification tests where items need to be regularly renewed).
- It allows dynamic recalibration of the model as new data is gathered.

The last two features are highly regarded by practitioners since test content is often subject to frequent updates which impose a strong burden for the maintenance of CAT test content.

Graph Model Considerations

What IRT lacks is the ability to make detailed cognitive assessment such as identifying specific concepts or misconceptions. In the original IRT, there is no provision for dividing the knowledge domain into different concepts that can be assessed individually, except by segmenting a large test into smaller ones, or by using what is known as multidimensional IRT (MIRT) models (Reckase, 1997; McDonald, 1997). But as we move towards MIRT, then some knowledge engineering effort is required to identify the dimensions and to classify items according to each of these dimensions. It becomes a graph model with multiple hidden nodes.

[4] Besides mastered and non-mastered, a third category is often used, *undecided*. In theory, any number of categories can be defined.

Our review of Bayesian student modeling revealed that the prevalent approach is to follow knowledge engineering techniques to build sophisticated graphical models with multiple levels of hidden nodes. Such models are often structured into a hierarchical decomposition of concepts into more and more specific skills, with items as leaf nodes. In some variants, misconceptions, multi-parents nodes, and sibling links can add yet more cognitive assessment and representation power to such structures. This is an essential feature of many intelligent learning environments that rely on fine grained student modeling.

However, this flexibility comes at the cost of modeling efforts to define the structure by domain experts, who must also be knowledgeable in Bayesian modeling. Beyond the structural definition, the problem of calibrating hidden node relations and nodes with multiple parent relations is paramount because of the lack of sufficient data cases (Jameson, 1995). Complex graph models often involve simplifications and approximations, such as leaky-AND/OR gates (Martin & Vanlehn, 1995; Conati et al., 2002) and weighted means (Millán & Pérez-de-la-Cruz, 2002), thereby weakening the validity and accuracy of the model.

As a consequence of the above obstacles, complex graph models leave little room for automation and its benefits. Although recent developments have shown that small networks of a few tens of nodes can be reliably derived from empirical data of a few thousand cases (Cheng, Greiner, Kelly, Bell, & Liu, 2002), this is still impractical in student modeling and the automated construction of a BN network remains a difficult problem that involves complex algorithms and considerable computing resources. In practice, heuristics and some degree of expert intervention are required for building a BN. With the exception of Vomlel (2004) who has used the combination of a network topology induction system with knowledge engineered adjustments to the structure, Bayesian student models do not allow automated model building. When used, empirical data serves the sole purpose of calibrating conditional probabilities, and yet, many also use subjectively estimated parameters.

Item Node Structures

Item node structures are particularly subject to the difficulties of using Bayesian graph models because the number of nodes can be large (e.g. in the French language test used for this study, we have 160 item nodes) and their structure is not as apparent as when dealing with concepts. Nevertheless, the theory of knowledge spaces (Falmagne, Koppen, Villano, Doignon, & Johannesen, 1990) states that items do have a structure and that it can be used for making knowledge assessments. But the obstacles to using a knowledge engineering approach and the amount of data required for the precise calibration of Bayesian networks make such an approach impractical.

We will see that POKS addresses these obstacles by reverting to binary relations, which allows calibration with small data sets, and using strong assumptions. That approach makes POKS amenable to algorithmic model construction and calibration. However, the issue of detailed cognitive assessment remains since concepts have to be included to provide fine grained assessment.

COMPARISON BETWEEN THE IRT-2PL MODEL AND THE POKS MODELING TECHNIQUES

The previous sections cover the general background theory and the issues that surround this study. We now describe in the next two sections the specific methods used for the experiment, namely IRT 2PL model and the POKS approach.

The IRT-2PL model

The “2PL” IRT model stands for “two parameter logistic” model. The two parameters in question are item *difficulty* and *discrimination*. As mentioned before, these parameters control the shape of the ICC function (Figure 1). The 1PL model drops the discrimination factor, whereas the 3PL adds the additional “chance factor” parameter that accounts for lucky guesses. Because chance factor is relatively small in our data and because its use does not necessarily lead to better results (see Baker, 1992), we use the 2PL model. The details of the 2PL model follows.

Let X_i be the examinee’s response to item i on a test of k items that can either be succeeded, $X_i = 1$, or failed, $X_i = 0$. Then, assuming item independence according to equation (2), the likelihood of a sequence of response values given an ability θ is:

$$P(\mathbf{X_k} \mid \theta) = \prod_{i=1}^{k} P(X_i|\theta) \tag{3}$$

where $\mathbf{X_k}$ is the vector of response values $X_1, X_2, \ldots, X_k$.

In the logistic IRT model, the probability of an examinee of ability level θ to answer item i correctly is:

$$P(X_i \mid \theta) = \frac{1}{1 + e^{-a_i(\theta - b_i)}} \tag{4}$$

where b_i is the difficulty parameter for item i, and a_i is its discrimination parameter. This function defines the ICC function of Figure 1. Note that a constant of 1.7 is often embedded in equation (4) as a factor of a_i to bring the shape of the ICC function closer to what is known as the *normal ogive model*, which was the prevalent IRT model until recently. This is not necessary for our purpose.

Equations (3) and (4) provide the basis of examinee ability estimation: it is possible to estimate the most probable ability level, θ, given the sequence of previous answers by maximizing the likelihood function in equation (3). This procedure is usually done by using a log likelihood function maximization model. Baker (1992) reviews the different algorithms that are used for this estimation procedure.

The POKS Bayesian modeling approach

We now turn to a short description of the POKS Bayesian modeling approach used for this study.

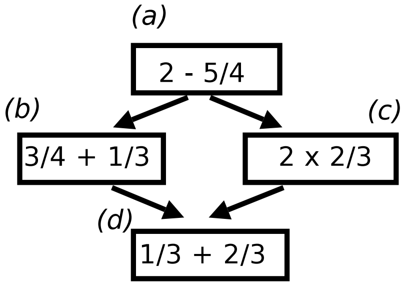

Fig.3. Simple knowledge structure example.

Item to item node structures and the theory of knowledge spaces

Probably the most distinctive characteristic of POKS is that it permits the inference of the structure among item nodes. However, it is not the sole attempt in this direction as Falmagne et al. (1990) and a number of followers (e.g. Kambouri, Koppen, Villano, & Falmagne, 1994) have worked towards using the structural characteristics of item-to-item structures to infer an individual's knowledge state. POKS also derives from this framework.

Item to item relations have their cognitive grounding in the theory of *knowledge spaces* (Falmagne et al., 1990; Kambouri et al., 1994) and they are termed as *surmise relations*. The meaning of such relation is essentially that we expect people to master these items in the reverse order of these relations.

Figure 3 illustrates such type of relations with a simple example. It can be seen that the example comprises of the following surmise relations: $a \rightarrow b \rightarrow d$ and $a \rightarrow c \rightarrow d$. However, no relation exists between b and c. For example, if a pupil succeeds item a, it will increase the estimated probability of success to items b, c, d. Conversely, failure to item d will decrease the estimated probability of success to items a, b, c. Finally, failure or success between items b and c will not affect the estimated probability of success to the node according to the *knowledge spaces* theory[5].

However, POKS does not strictly conform to the knowledge spaces theory because it uses partial orders such as Figure 3, whereas *knowledge structures* use AND/OR graph. The difference is that partial orders define possible knowledge states closed under union *and* intersection whereas AND/OR graphs define possible knowledge states closed under union only. Indeed, defining the knowledge state of an individual as a subset of a global set of knowledge items, Falmagne and his colleagues established that the set of possible knowledge states from global set of items is constrained by closure under union: if we join two individuals' knowledge state, this is also a possible knowledge state (for details, see Falmagne et al., 1990). If, for example, we define X_b and X_c as two items that test different methods of solving a problem and that any one of these methods can be used in solving a third item X_a (at least one must be used), this would be reflected in knowledge structures theory as an OR relation binding the three nodes and clearly expressing the alternative found in the relation. It is also clear that the intersection of two individuals, each mastering a single alternative method between X_b and X_c, would yield an invalid

[5] Note that this is not the case for the IRT theory, nor of the Bayesian modeling techniques reviewed in this paper which links every test item to a global ability.

knowledge state: Someone who masters X_a but none of X_b and X_c (we ignore probabilistic issues here). In POKS, we would likely find weak surmise relations $X_a \rightarrow X_b$ and $X_b \rightarrow X_a$, capturing some of the information but not as accurately as with an OR relation.

Nevertheless, because partial orders do capture to a large extent the constraints on possible knowledge states and because the probabilistic nature of POKS makes it more flexible and robust to noise, the use of partial orders remains a powerful means of making knowledge assessment. Moreover, because OR relations are tertiary or higher n-ary relations, they impose larger data sets to discover and are thus more limited in their applications.

Local independence

Another characteristic of POKS is that it makes the assumption of local independence among evidence nodes. In POKS, we essentially make the assumption that we can limit the modeling solely to binary conditional probability relations. More formally, we make the assumption that for any node X having parents $pa(X) = \{X_{p1}, \ldots, X_{pn}\}$, all parents' influence is independent of each other:

$$P(X|X_{p1}, \ldots, X_{pn}) = \prod_{i}^{n} P(X|X_{pi}) \tag{5}$$

Although this assumption is obviously violated in most contexts, the question of whether it is leads to significant errors is an empirical question that will be assessed and discussed further[6]. The great benefit of making this assumption is that it allows the induction of the network from a very small number of data cases. Only the analysis of binary relations data is needed to create the model. In the current experiment, less than 50 data cases were used to build the model.

POKS network induction

Knowledge structures such as the example in Figure 3's example are learned from data. The POKS graph model and induction technique is briefly reviewed here.

Nodes:

As mentioned above, and akin to other user modeling graph models, POKS structures can include nodes that represent concepts or test items. However, for the purpose of comparing IRT and POKS, the nodes are limited to representing test items. There are no other types of nodes; each node is a test item, and each test item is a node. All items have equal weight for this experiment.

Each node, X_i, is assigned a probability that represents an examinee's chances of mastery of that item, $P(X_i)$. Contrary to the IRT model, $P(X_i)$ is not a function of θ, the ability level. It is a direct function of the probability of other items from which it is linked with. The details of how to compute $P(X_i)$ in POKS is described in the *Item probability update* section.

Relations:

[6] Note that the local independence assumption is also an issue in CAT (Mislevy & Chang, 2000).

Relations in POKS have the same meaning as knowledge spaces' *surmise* relations; they indicate the (partial) order in which people learn to master knowledge items (see the section on *Item to item node structures*). Although *surmise* relations are different from causality relations found in Bayesian networks, they allow the same type of inferences[7]. For example, let X_a and X_b be two items in an item bank, a relation $X_a \rightarrow X_b$, means that observing an examinee succeed in item X_a will increase the estimated probability of success in item X_b by a certain amount. Conversely, a failure in item X_b will decrease the estimated probability of success to item X_a.

Networks structure:

In accordance with the assumption of local independence, the network construction process consists in comparing items pairwise to look for a relation. To determine if there is a directed link, $X_a \rightarrow X_b$, the three following conditions must hold:

$$\begin{aligned} P([P(X_b|X_a) \geq p_c] \mid D) > &\quad (1 - \alpha_c) \qquad (6) \\ P([P(\neg X_a|\neg X_b) \geq p_c] \mid D) > &\quad (1 - \alpha_c) \qquad (7) \\ P(X_b|X_a) \neq P(X_b) &\quad (p < \alpha_i) \qquad (8) \end{aligned}$$

where:

p_c is the minimal conditional probability for $P(X_b|X_a)$ and $P(\neg X_a|\neg X_b)$; an single value is chosen for the test of all relations in the network, generally 0.5.

α_c is the alpha error of the conditional probability tests (6 and 7); it determines the proportions of relations that can erroneously fall below p_c; common values range from 0.2 and 0.5.

$p < \alpha_i$ expresses the alpha error tolerance of the interaction test (8).

D is the joint frequency distribution of X_a and X_b in the calibration sample. This joint distribution is a 2×2 contingency table with four frequency numbers, $\{x_{ab}, x_{a\neg b}, x_{\neg ab}, x_{\neg a \neg b}\}$, representing the number of examinees in the sample data broken down into these four situations:

1. x_{ab}: success for X_a and X_b
2. $x_{a\neg b}$: success for X_a and failure for X_b
3. $x_{\neg ab}$: failure for X_a and success for X_b
4. $x_{\neg a\neg b}$: failure for X_a and X_b

The first condition (inequality (6)) states that the conditional probability of a success for X_b given a success for X_a must be above a minimal value, p_c, and that we can derive such conclusion from a sample data set, D, with an error rate smaller than α_c. The second condition (inequality (7)) is analogous to the first and states that the probability of failure for X_a given a failure for X_b must be greater than p_c, with a maximal error rate of α_c given distribution D.

[7] In fact, causality also has the property of ordering events in time, and it is a non trivial philosophical endeavor to determine that it has any other property!

These first two conditions are computed from the cumulative Binomial distribution function. In inequality (6), the value of $P([P(X_b|X_a)]|D)$ is obtained by the summation of the Binomial probability function for all distributions where $x_{a\neg b}$ are less than the number actually observed in D, that is:

$$\begin{aligned} P([P(X_b|X_a)]|D) &= P(x \leq x_{a\neg b} \,|\, X_a) \\ &= \sum_{i=0}^{x_{a\neg b}} \text{Bp}(i, x_a, p_c) \\ &= \sum_{i=0}^{x_{a\neg b}} \binom{x_a}{i} p_c^{[x_a - i]}(1 - p_c^i) \end{aligned}$$

where $x_a = x_{ab} + x_{a\neg b}$. The conditional probability of the second condition (inequality (7)) rests on the same function but uses $\text{Bp}(i, x_{\neg b}, p_c)$ in place of $\text{Bp}(i, x_a, p_c)$.

The third condition (inequality (8)) is an independence test and it is verified by a χ^2 distribution test on the 2×2 contingency table of distribution D:

$$P(\chi^2) < \alpha_c$$

For small samples, the independence test used is replaced by the Fisher exact test.

The choice of value for the p_c indicates the strength of the *surmise* relations we want to keep. For example, if the order in which one learns to master two items is highly constrained, in accordance with the theory of *knowledge spaces*, then we would expect to find that $P(B|A) \approx 1$ for a strong *surmise* relation $X_a \to X_b$. The value of p_c represents the lower limit for which we accept a *surmise* relation. The choice of a value is somewhat arbitrary, but we generally use $p_c = 0.5$ in our experiments.

The two values α_c and α_i represent the alpha error we are willing to tolerate when concluding the corresponding tests. For very small samples, these values can be as high as 0.5 in order to keep as many relations as possible. In our experiments they are set between 0.2 and 0.1.

Item probability update

When an item's probability of mastery in the network changes, either through observation or through a change in the probability of a neighboring node, evidence is propagated through the connected items in the network. If the probability increases, the update will follow links forward, whereas if the probability decreases, the update will follow links backward. We use the algorithm for evidence propagation from Giarratano and Riley (1998). This algorithm is consistent with the Bayesian posterior probability computation in single layered networks and corresponds to the posterior probability update. However, for multilayered networks, in which indirect evidence gets propagated (transitive evidence from non-directly connected nodes), an interpolation scheme is used. This is explained in a numerical example given later.

For computational convenience, the algorithm relies on two odds ratios: the *likelihood of sufficiency* and the *likelihood of necessity* respectively defined as:

$$LS_{a\to b} = \frac{O(X_b|X_a)}{O(X_b)} \tag{9}$$

$$LN_{a \rightarrow b} \;\; = \;\; \frac{O(X_a|\neg X_b)}{O(X_a)} \tag{10}$$

where $O(X)$ is the odds function, $P(X)/Q(X)$ (where $Q(X) = P(\neg X) = 1 - P(X)$), and $O(X|Y)$ is the conditional form, $P(X|Y)/Q(X|Y)$.

It follows that if we know X_a to be true (i.e. $P(X_a) = 1$), then the probability of X_b can be updated using this form of equation (9):

$$O(X_b|X_a) = LS_{a \rightarrow b}\, O(X_b) \tag{11}$$

and conversely, if X_a is known false, then:

$$O(X_a|\neg X_b) = LN_{a \rightarrow b}\, O(X_a) \tag{12}$$

The update process recursively propagates forward using equation (11) when a node's probability increases, and backward using equation (12) when it decreases.

In accordance with the local independence assumption in equation (5), it follows that the odds ratios are combined as the product of the LS of each parent that is observed:

$$O(X_j|pa_o(X_j)) = O(X_j) \prod_{X_i \in pa_o(X_j)} LS_{i \rightarrow j} \tag{13}$$

where $pa_o(X_j)$ are the observed parents of node X_j and $O(X_j)$ is the initial odds ratio. Conversely, the LN odds ratios are also combined for the children nodes:

$$O(X_k|ch_o(X_k)) = O(X_k) \prod_{X_i \in ch_o(X_k)} LN_{k \rightarrow i} \tag{14}$$

where $ch_o(X_k)$ are the observed children of node X_k. We emphasize again that this strong assumption is surely violated in most contexts, but it greatly simplifies node updates by relying on functional computations (as opposed to the computations required for optimizing a system of equations) and on the network's Markovian property; only the network's current state is sufficient to make future predictions. The impact of this assumption's violation will be assessed in the experimental evaluation.

Evidence propagation directionality

The evidence propagation scheme is unidirectional in the sense that if a node's probability increases, no backward propagation is performed, and, conversely, no forward propagation is performed when a node's probability decreases. This may look as a breach in standard Bayesian theory since posterior updates can occur in both directions. In fact, it is not. It follows from POKS principle of pruning non-significant posterior updates relations with the statistical tests (6), (7), and (8). Let us illustrate this with a simple example. Assume the following two question items:

a : Examinee is able to solve for x: $\frac{3}{2x} \times \frac{7}{4} = \frac{3}{8}$

b : Examinee is able to find the answer to $\frac{3}{7} \times \frac{7}{4} =?$

The POKS induction algorithm would readily derive $a \rightarrow b$ from a data sample taken from the general population on these two items, indicating that it is worth updating b's posterior probability if we observe a. However, the converse relation $b \rightarrow a$ would probably fail the statistical tests for inequalities (6) and (7), indicating that the inverse relation is not strong enough. Indeed, it is fairly obvious that a success for item b does not significantly increase the chances of success of a because the latter involves algebra and is significantly more advanced than the former. However, if we replace a with an item of closer difficulty to b, such as:

a : Examinee is able to find the answer to $\frac{4+8}{11} \times \frac{11}{12} =?$.

then we would probably also derive $b \rightarrow a$. The result would be a symmetric relation (probably with different LN and LS values for $a \rightarrow b$ and $b \rightarrow a$). In that case, a probability increase or decrease in any node would affect the other node's probability in accordance with Bayes posterior probability update, and propagation would be bi-directional.

When relations are symmetrical, $X_b \rightarrow X_a$ and $X_a \rightarrow X_b$, cycles involving two nodes are created. There are two solutions to this problem, the first of which consists in grouping symmetrical nodes into a single one. A second solution, adopted for this study, is simply to keep symmetrical relations but to stop the propagation of evidence once a node has already been visited during a single propagation run. This is a standard procedure in message propagation and constraint programming systems.

Numerical example

Let us illustrate numerically the evidence propagation with an example. Assume the following relations hold:

$$a \rightarrow c, \; b \rightarrow c$$

and that in our sample we find:

$$P(X_c) = 0.3, \; P(X_c|X_b) = 0.6, \; P(X_c|X_a) = 0.9$$

It follows from the above equations that observing X_a first (i.e. $P'(X_a) = 1$)[8] would bring $P'(X_c) = 0.9$, which corresponds to the value of the sample's observed conditional probability $P(X_c|X_a)$. Further observing X_b would bring $P''(X_c) = 0.969$, which corresponds to $P(X_c|X_b, X_a)$. Inversion of the order of observation would bring instead $P'(X_c) = 0.6$ after observing X_b (i.e. $P(X_c|X_b)$) and $P''(X_c) = 0.969$, as expected (i.e. $P(X_c|X_b, X_a)$).

Although odds are used in the algebra for computing the posterior probabilities, it is equivalent to using the standard Bayes formula for obtaining the posteriors given the observation $X = 1$ or $X = 0$. However, when the probability of a node increases or decreases, an interpolation scheme is used to further propagate evidence.

[8] We use the notation $P'(X)$ to represent an updated probability and drop the conditional form, $P(X|evidence)$, to better emphasize the stages of updating. $P'(X)$ is the value of $P(X)$ after the first stage of updating, whereas $P''(X)$ is the value after the second stage.

When the probability of one of a node's parent nodes changes by some value, without actually being observed and thus reaching the value of 1 or 0, two interpolation formulas are used to update this node's probability. Assuming a relation $a \rightarrow b$, and an increase in $P(X_a)$ of Δ_a (i.e. $P'(X_a) = P(X_a) + \Delta_a$), where $P(X_a)$ represents the probability before the update and $P'(X_a)$ the probability after the update), then the value of $P'(X_b)$ is given by:

$$P'(X_b) \quad = \quad P(X_b|X_a) + [P(X_b|X_a) - P(X_b)]\frac{P'(X_a) - P(X_a)}{P(X_a)}$$

where $P(X_b)$ is the probability of X_b before the update.

Following $a \rightarrow b$ in the backward propagation direction and assuming a decrease $P(X_b) - P'(X_b) = \Delta_b$, the updating formula is:

$$P'(X_a|\neg X_b) \quad = \quad P(X_a|\neg X_b) + [P(X_a) - P(X_a|\neg X_b)]\frac{P'(X_b)}{P(X_b)}$$

This interpolation method is a simple approximation of $P(X|E_1, E_2)$, where $E_1 \rightarrow X$ and $E_2 \rightarrow E_1$ are directly linked, but $E_2 \rightarrow X$ are not. Its validity for the field of CAT is a question we investigate empirically in this study. More details about the interpolation method can be found in Giarratano and Riley (1998).

Item selection

Common to both the IRT and the BN approaches is the problem of choosing the next question in order to minimize the number of questions asked. As discussed in the *item selection* section, there exists a number of measures to choose the most informative item. We used two for the purpose of this study: The Fisher information and the *information gain* criteria. They are described below.

Fisher information

One of the most widely known criteria for choosing the next item is the Fisher information (Birnbaum, 1968). It is an indicator of the sensitivity of an equation, such as a likelihood, to changes in a given parameter. It is a widely used metric in statistical modeling. In the case of IRT, the principle is to identify the item which is most likely to induce a change in the estimated θ.

In the two parameter IRT model, the Fisher information for item i is given by:

$$I_i(\theta) = a_i^2 \frac{e^{a_i(\theta - b_i)}}{[1 + e^{a_i(\theta - b_i)}]^2} \tag{15}$$

where a and b are the two parameters of the ICC function (see the section entitled *IRT-2PL model*). This measure will essentially identify the item whose inflexion point of the ICC curve (Figure 1) is closest to the estimated θ and with the highest discrimination value a.

In POKS, θ is computed from a measure of the estimated mastery level. That measure, m, corresponds to average probability over all k items:

$$m = \frac{\sum_i^k P(X_i)}{k}$$

Note that we could also have used the expected item success rate as an alternative, but the current measure is more sensitive as it discriminates between an item with a probability of success $.49$ and another with probability $.01$.

The value of m varies on a scale $[0, 1]$, whereas θ is on the scale $[-\infty, +\infty]$. To bring m onto the θ scale, we apply the `logit` transformation, with parameters a and b:

$$\theta_m = \mathtt{logit}(m)/a + b = \mathtt{log}(\frac{m}{1-m})/a + b$$

For the IRT model, the value of θ is computed by maximizing equation (3) using maximum likelihood estimation.

Information gain

The second approach to item selection we investigate is the *information gain* approach. The principle of this approach is to choose the item that will maximize the expected reduction of entropy of the test. This is explained below.

The entropy of a single item X_i is defined as:

$$H(X_i) = -[P(X_i)\mathtt{log}(P(X_i)) + Q(X_i)\mathtt{log}(Q(X_i))] \tag{16}$$

where $Q(X) = 1 - P(X)$. The entropy of the whole test is the sum of all individual item's entropy:

$$H_T = \sum_i^k H(X_i)$$

If all item probabilities are close to 0 or 1, the value of H_T will be small and there will be little uncertainty about the examinee's ability score. It follows that we minimize this uncertainty by choosing the item that maximizes the difference between the current test entropy, H_T, and the entropy after the item's response, H'_T. The expected value of the whole test entropy after a response to item X_i is given by:

$$E_i(H'_T) \;=\; P(X_i)H'_T(X_i\!=\!1) + Q(X_i)H'_T(X_i\!=\!0)$$

where $H'_T(X_i = 1)$ is the entropy after the examinee answers correctly to item i and $H'_T(X_i = 0)$ is the entropy after a wrong answer. We then look for the item that will have the maximum difference:

$$\max_i \quad [H_T - E_i(H'_T)]$$

EXPERIMENTAL EVALUATION OF THE APPROACHES

The POKS approach is compared to a 2-parameter IRT model (see section *IRT-2PL model*). Furthermore, two item selection procedures are investigated, namely the and Fisher information the *information gain* approaches. A random item selection procedure is also reported for baseline result comparison.

Methodology

The performance comparison rests on the simulation of the question answering process. For each examinee, we simulate the adaptive questioning process. The answers given by the examinee during the simulation are based on the actual answers collected in the test data. The classification by the IRT-2PL and POKS approaches after each item response given is then compared to the actual examinee score in the test data. An examinee is classified as *master* if the estimated score is above a given *cut score*, θ_c, and *non-master* otherwise. This cut score is expressed as a percentage, but recall that transformation of a percentage to IRT's θ score that generally lies between $[-4, +4]$ is done through the logit transformation (see the *Fisher information* section).

The items responded by the examinee are taken as the *true* ability score for the part of the test they cover, and only the remaining items are estimated. The overall ability estimate is thus a weighted sum of the probability of success to already *responded* items and *estimated* items. That is, if I_r is the set of items responded and I_e is the set of items estimated, the examinee's estimated score, S, is:

$$S = \frac{\sum_{X_i \in I_r} X_i + \sum_{X_j \in I_e} \hat{X}_j}{n} \tag{17}$$

where X_i is 1 if the corresponding response to item i is a success and 0 otherwise, and where $\hat{X}_j$ is 1 if the estimated probability of success, $P(X_j)$, with the respective method used, POKS or IRT-2PL, is above 0.5 and 0 otherwise. Recall that in the IRT 2PL model, the probability of success to an item is given by equation (4), whereas in POKS it is computed through the propagation of evidence as explained previously. This procedure results in a 100% correctly classified examinees after all test items are observed[9].

Test data

The simulations are made on two sets of data: (1) a 34 item test of the knowledge of UNIX shell commands administered to 48 examinees, and (2) a 160 item test of the French language administered to 41 examinees. The first test was designed by the first author and it assesses a wide range of knowledge

[9] This validation procedure rests on the fact that we do not know the actual ability state of an examinee apart from the test results. Indeed, contrary to a frequently used approach that consists in generating test response data cases from Monte Carlo simulations, we use real data to validate the models. This procedure has the obvious advantage of having good ecological validity. However, it leaves us with the epistemological position of having the test data as the sole indicator of examinee ability. Performance results, then, should be interpreted as the ability of the models to *predict examinee score* for the given test. If we assume that a test is a true reflection of ability, then we can extend the interpretation of the models' performance as a measure of their accuracy to predict examinee ability.

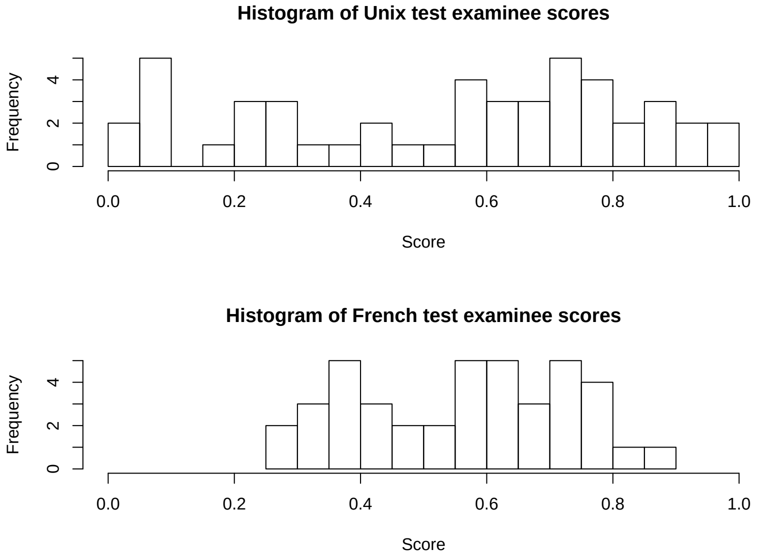

Fig.4. Histogram of examinee scores for each test.

of the UNIX commands, from the simple knowledge of ‘cd’ to change directory, to the knowledge of specialized maintenance commands and data processing (e.g. ‘awk’, ‘sed’). The second is a test used by the Government of Canada. It is designed by professionals and covers many aspects of the language in question and a wide range of language skills.

Mean scores for the UNIX and French language tests are respectively 53% and 57%. Figure 4 illustrates the dispersion of scores for each test. A wide distribution of scores is necessary for the proper calibration of both POKS and the IRT-2PL model.

To avoid sampling bias error, all calibrations of the models’ parameters are done on $N - 1$ data cases; we remove from the data set the examinee for which we conduct the simulation. As a result, simulations are conducted with parameters calibrated from 47 data cases for the Unix test and 40 data cases for the French language test.

Parameters estimation

The *discrimination* and *difficulty* (respectively parameters a and b in equation (4)) were estimated with a maximum log-likelihood estimator package of the R application (Venables, Smith, & the R Development Core Team, 2004) over the two data sets. These same parameters are also used in equation (15) for the choice of the next item with both the POKS and the IRT-2PL approaches.

Table 1

Graph statistics averaged over all $N-1$ structures.

	Unix graph	**French language graph**
Total number of relations	587	1160
Symmetric relations	252	131
Transitive relations	229	668
α_c	0.25	0.10
p_c	0.5	0.5

Graph structures and statistics

Statistics on the graph structures inferred are given in table 1. The number of relations reported represent the average over the 48 and 41 networks that were used for the simulations (one per simulation to avoid the over-sampling bias). Note that symmetric relations are in fact two directed relations between a pair of nodes (dividing the numbers by two gives the actual individual symmetric relations). Note also that, when counting transitive relations, groups of nodes linked through symmetric relations are merged into one single node[10], to avoid cycling. The numbers represent the transitive relations actually induced by the algorithm (not the relations that can be derived through transitivity).

The minimal conditional probability, p_c, for both tests networks is the same, 0.5. The values for α_c and α_i are 0.25 for the Unix data set and 0.10 for the French one. The choice of $\alpha_c = 0.10$ for the French language test proved to be more reliable during the preliminary testing. However, values of α_c ranging from 0.2 to 0.5 showed little effect on the results for the Unix data set, but performance degradation started to appear around $\alpha_c = 0.1$.

Computational resources

Computational resources for building the graph structure and performing inferences is often an issue for operational systems and thus we report some indicators here. For our test data, time for constructing a graph structure with the Unix and French language data set is very fast: less than 10ms on a standard 1.5Ghz PC. Inferences for CAT is also fast. We find that a full cycle involving (1) the update of item probabilities and (2) determining the next question to ask, varies from 0.03ms for the Unix test with the Fisher information condition, to a much longer 106ms for the French language test under the *information gain* condition. The *information gain* condition is much slower because it involves simulating correct and wrong responses to every other test item to find the expected entropy. Moreover, the high number of connections in the French language network significantly affects the time to compute the entropies for the *information gain* technique.

[10] Merging nodes of symmetric relations into one is only for the purpose of counting transitive relations and not for performing inferences.

Performance metrics

Measuring the performance of each approach is based on a simple metric: the proportion of correctly classified examinees after each number of responses to test items. Classification of a single examinee is determined by comparing the examinee's estimated score, S (equation (17)), with the passing score, θ_c.

The percentage of correctly classified examinees is reported as a function of the number of test item responses given. Figure 5 illustrates an example of a performance graph. The curve starts at 0 item, i.e. before any items are given, at which point we use the samples' average to initialize the probabilities of each test item. Each examinee will thus start with an estimated $\hat{\theta} = \overline{X}$, the sample's average score in percentage points. If the sample average is above θ_c, all examinees will be considered *master*, otherwise they are considered *non-master*. As a consequence, the performance at 0 item generally starts around 50% when the cut score is around the sample's average, and gradually reaches 100% at the end of the test, when all items are observed. As the cut score departs from the average, the 0 item initial performance (or "blind score") increases and eventually reaches 100% if everyone is above or below this score in the sample. For example, at a cut score of 80% this initial score is $40/42$ for the French language test because only two examinees score above this level and we start with the estimate that everyone is a *non-master*.

The diagonal line in Figure 5 represents a baseline performance used in measuring a global score, G (see below).

Thus, region C of Figure 5 represents a linear approximation of the "given" (i.e. the proportion of examinees that are are correctly classified due to gradual observation of responses of test items), region B represents the "correct inferences" (i.e. the proportion of examinees correctly classified by the inference method), and region A represents "wrong inferences" (i.e. the proportion that are still incorrectly classified).

Besides graphically reporting the classification results, a single scalar metric, G, is defined for characterizing the performance over a complete simulation run. It corresponds to the ratio of surfaces $B/(A + B)$ in Figure 5 and is computed by:

$$G = \sum_{i=1}^{k} \frac{C_i - Ce_i}{n - Ce_i} \qquad (18)$$

where n is the number of examinees, k the number of items, C_i is the number of correctly classified examinees after i number of item responses (the line with black circles), and Ce_i the expected number of examinees correctly classified by sole observation of test items (i.e. the diagonal line in the performance figures). G values can range from $\frac{-k}{2n/[(nCe_0)-1]}$ (where Ce_0 is the number of correctly classified examinees before any response is given), to 1. A value of 1 represents a perfect classification throughout the simulation, a value of 0 indicates no gain over observation only, and a negative value indicates a worse classification than that obtained by combining the 0 item initial classification with the given responses.

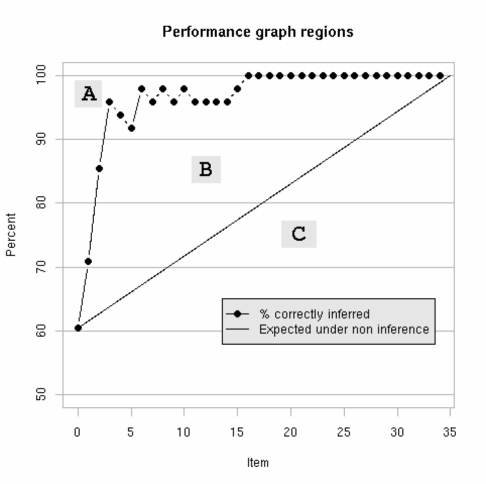

Fig.5. Regions of a performance graph. The global score metric, G (equation (18)), represents a ratio of surface regions $B/(A + B)$

Results

Simulations at $\theta_c = 60\%$

The simulation results for the cut score $\theta_c = 60\%$ are summarized in Figure 6 for the Unix and French Language tests. They show the number of correctly classified examinees as a function of the number of items asked. For better visibility, the French language test data points are plotted every 4 items.

Both the *information gain* and the Fisher information item selection strategy are reported for the POKS model. However, for IRT-2PL approach, only the Fisher information function is given because of limitations with the IRT simulation software we are using. The Fisher information is the most widely used item selection strategy for IRT. We refer the reader to Eggen (1998) for a comparison of different item selection strategies with IRT.

The simulation shows that both POKS and IRT-2PL approaches yield relatively good classification after only a few item responses, especially considering the low number of data cases used for calibration. In the UNIX test, all approaches reach more than 90% correctly classified between 5 and 10 item responses. However, for the French language test, only the POKS-*information gain* and POKS-Fisher information approaches stays above 90% correct classification after about 20 items, whereas the IRT approach requires about half of the 160 test items to reach and stay above the 90% score.

At this 60% passing score, we can conclude that the POKS-*information gain* approach performs better in general than the two others but, as we see later, this advantage is not maintained for all different cut scores.

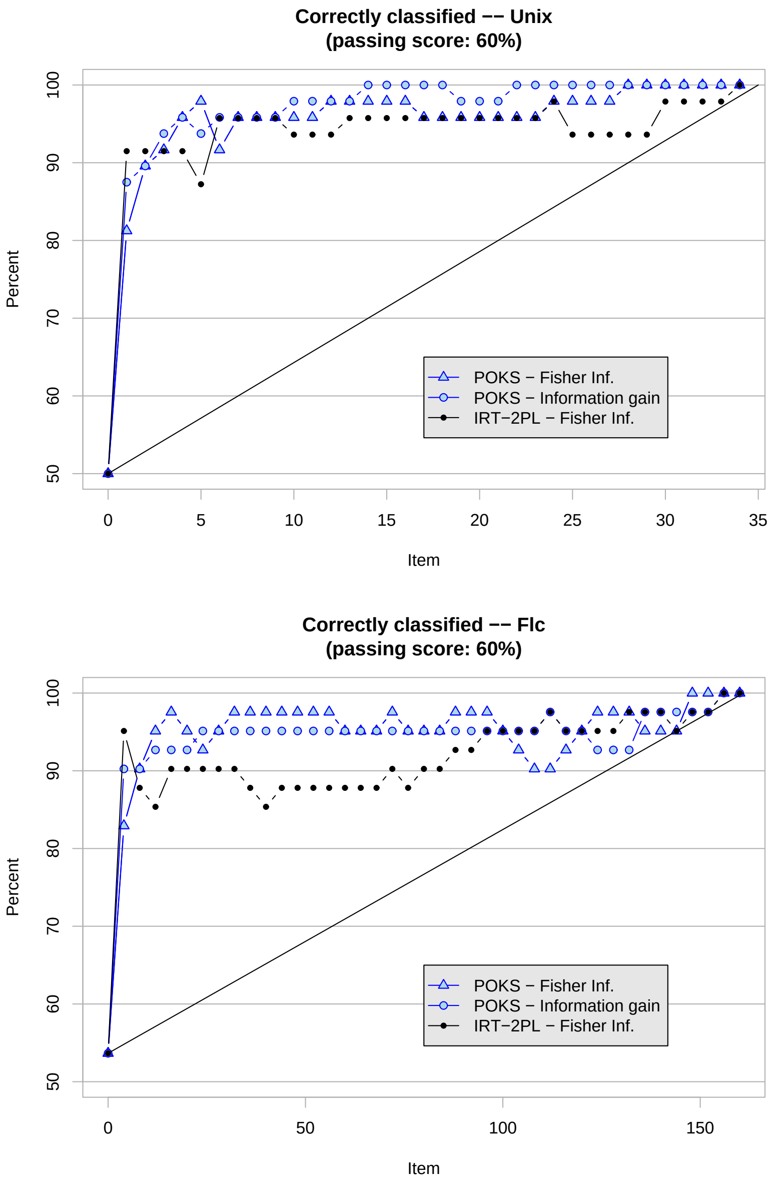

Fig.6. Results from the Unix (top) and French language (bottom) tests comprised respectively of 48 and 160 items. The percentage of correctly classified examinees, averaged over 48 simulation cases for the Unix test and 41 for the French language one, are plotted as a function of the number of item responses. Passing score is 60%.

Table 2
Performance comparison of the G metric (equation (18)) under different conditions of cut cores and α values (inequalities (6) and (7)). Three item selection techniques are reported for the POKS approach (*information gain, Fisher information,* and random item selection with a 95% confidence interval) , whereas only the *Fisher information* technique is reported for the IRT framework, which is the most commonly used.

	IRT	**POKS**				
θ_c	**Fisher Inf.**	**Fisher Inf.**		**Inf. Gain**		**Random (95% c.i.)**
UNIX test		$\alpha = .25$	$\alpha = .15$	$\alpha = .25$	$\alpha = .15$	$\alpha = .25$
50%	0.93	0.85	0.89	0.86	0.84	0.75(± 4.4)
60%	0.81	0.92	0.89	0.86	0.84	0.77(± 4.8)
70%	0.75	0.80	0.81	0.68	0.67	0.50(± 7.1)
average	0.83	0.86	0.86	0.80	0.78	0.67(± 7.1)
French Language test		$\alpha = .10$	$\alpha = .15$	$\alpha = .10$	$\alpha = .15$	$\alpha = .10$
50%	0.81	0.72	0.80	0.80	0.85	0.64(± 2.0)
60%	0.68	0.78	0.79	0.74	0.79	0.57(± 5.7)
70%	0.69	0.60	0.60	0.83	0.74	0.48(± 4.9)
average	0.73	0.70	0.73	0.79	0.79	0.54(± 7.1)

Performance under different θ_c and item selection strategies

To gain an idea of the general performance of POKS under different conditions, we investigate the following variations:

- Different cut score, from 50% to 70%[11];
- Item selection strategies, including a random selection of items,
- Two different values of the α_c and α_i parameters for inequalities (6) and (7) (we set $\alpha = \alpha_c = \alpha_i$). One set at $\alpha = 0.15$ for all conditions, and another one tailored for each test and corresponding to the graphs of Figure 6.

Table 2 summarizes the results of the simulations under these different conditions. The random selection represents the average of 9 simulation runs for each cut score. We use the G metric for reporting the performance of a whole simulation, from the first to the last test item, into a single scalar value.

The G metric at the 60% level reflects what is apparent in the graphs of Figure 6, namely that POKS has a slight gain over IRT and that all approaches perform better for the UNIX test than the French language test. However, the POKS advantage is not systematic for all cut scores and across the two item selection techniques. The averages of cut scores across the tests suggest a relatively similar performance between POKS and the IRT model. The average score advantage is inverted between POKS-Fisher information and POKS-*information gain*, but exploratory work (not reported here) with different

[11] Scores above 70% and below 50% are not reported because the large majority of examinees are correctly classified initially (as one can tell from Figure 4) and little performance gain is possible (at least for the French test). Reporting scalar values within these ranges becomes misleading.

statistical parameters for the statistical tests (inequalities (6), (7), and (8)) indicates that this inversion is not systematic. All methods perform better than a random selection of items, as expected.

There is a noticeable decrease of performance for POKS at the 70% cut score where the Fisher information method score drops to 60% for the French test, and also drops for the UNIX test, but this time over the *information gain* method. This suggests that POKS may suffer weaknesses at boundary conditions. We link these results to a known problem with POKS that is further discussed in the discussion.

Question predictive accuracy

The comparison of IRT and POKS is also conducted at the question level. In the previous sections, we assessed how accurate each method is at classifying examinees as *master* or *non master* according to some passing score. The question predictive evaluation is a more fine grained assessment of the ability of each method to predict the outcome of each individual question item. In principle, the ability of an approach to predict individual item outcome offers a means for detailed student assessment, provided that individual skills and concepts can be related to specific items.

The measure for the question accuracy score is relatively simple. It consists in the ratio of correctly predicted item outcome and it is reported as a function of the number of items administered. For both methods, the probability of success of an item, $P(X_i)$, is continuously reassessed after each item is posed. If that probability is greater than 0.5, then the predicted outcome is for that item is a correct response, otherwise it is an incorrect response. Predicted responses are then compared with real responses for measuring their accuracy. Once an item is administered, the predictive accuracy score is considered 1 for that item and, as a consequence, the question predictive ratio always reaches 1 after all items are administered. All items are treated with equal weights.

Figure 7 reports the question predictive accuracy score for both tests. Only POKS *information gain* approach was investigated for this experiment. In addition to the two approaches, IRT-2PL and POKS, a third line is also displayed, “Fixed”. It represents the score for the simple method of choosing the most uncertain item remaining, i.e. the item whose sampled success ratio closest to 0.5. This method is non-adaptive; the sequence of items is fixed for all examinees. It serves as a baseline comparison. We note that the IRT approach starts at a lower score than the other two. This is due to the fact that the items probabilities, $P(X_i)$, is computed from the initial θ and that value turns out to be less accurate than taking the initial probabilities calibrated from the sample.

The standard deviations of the question predictive accuracy ratio is given in Figure 8. They are also reported as a function of the number of items administered and for the three corresponding curves of Figure 7.

The obvious finding is that POKS clearly outperforms IRT in the French language test, whose performance does not even match that of the fixed sequence method. However, it is not significantly better than the fixed sequence one. For the Unix test, POKS advantage over IRT is only apparent before the 10th item, but it does perform well above the fixed sequence, contrary to the French test.

These results confirm that the French test does not lend itself as well to adaptive testing as does the Unix test. This could be due to the smaller sample size (41 vs. 48) and the smaller sampling distribution (see Figure 4). The wider is the range of abilities, the easier it is to assess someone’s knowledge from

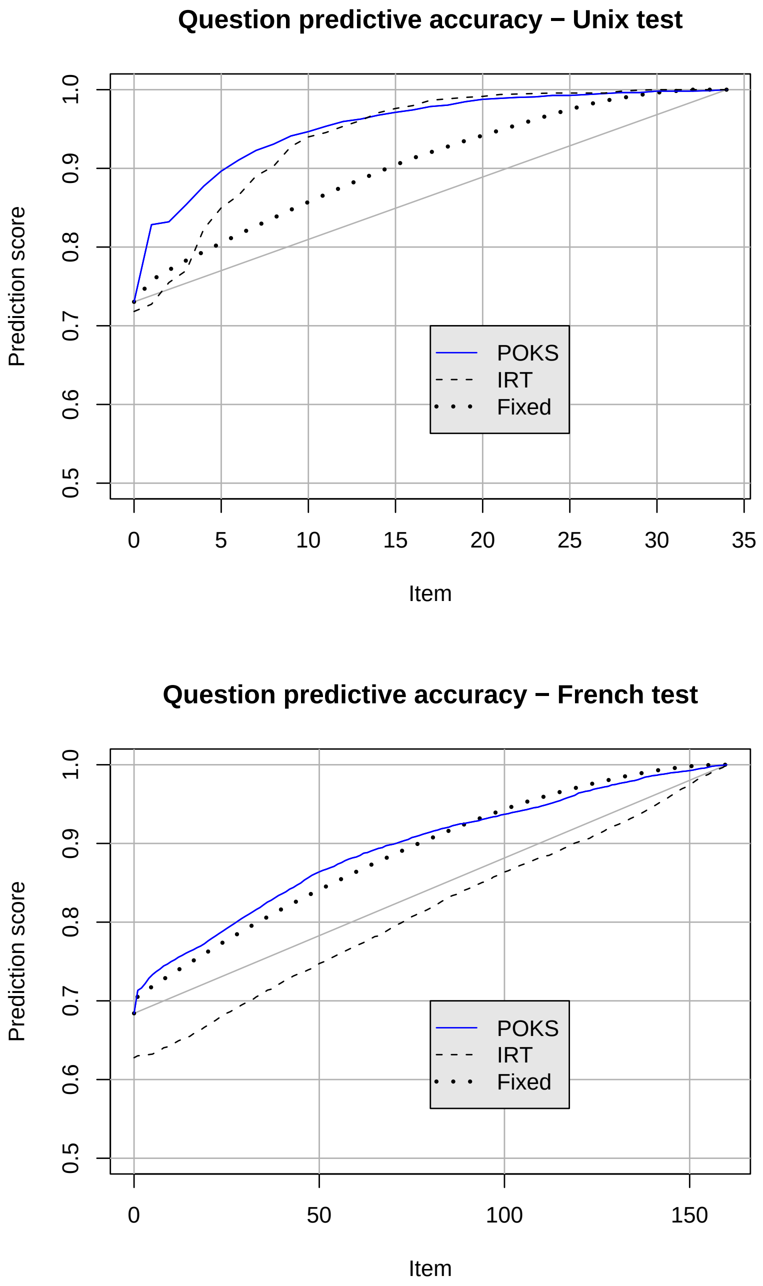

Fig.7. Individual test item prediction mean results from the Unix (top) and French language (bottom).

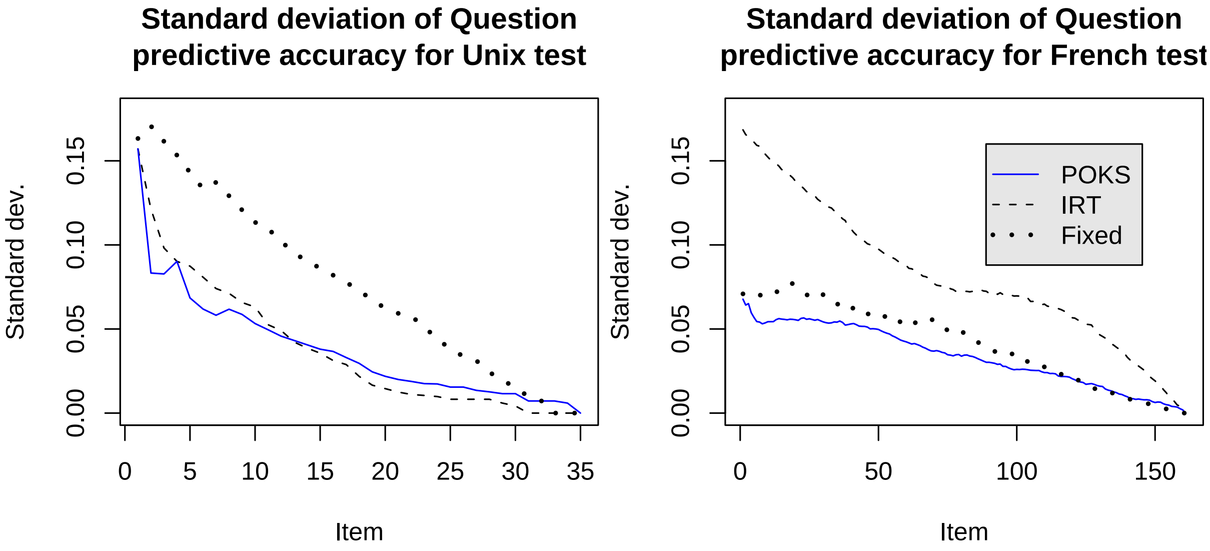

Fig.8. Standard deviations of each test as a function of the number of test item administered.

that sample[12]. It can also stem from the lower average item discrimination values (medians of 3.3 vs 1.5 for the UNIX and French tests respectively).

The low performance of IRT for the French test is not necessarily a surprise since IRT never claims to lend itself to fine grained and multidimensional skill assessment. However, it is a surprise that it *can* provide a good performance for the UNIX test, possibly because that test is more uni-dimensional than the French test, and also because the item success ratio distribution has a wider range. Obviously, an interesting followup would be to verify if a MIRT approach could yield better results for that test.

Nevertheless, the comparison does demonstrate that POKS has the potential of providing more fine grained assessment, if we assume that question predictive accuracy is more fine grained. For example, by segmenting tests items according to different skills, then individual question item prediction could provide useful information on individual skills. More investigation is required to confirm this hypothesis but, this is an early indication that supports it.

DISCUSSION

The comparison of the POKS approach with the IRT-2PL one shows that they both can perform correct classification of examinees under different tests and passing scores, although their performance differs according to that passing score. However, their ability to predict individual question item outcome varies considerably between themselves and also between tests. POKS can predict item outcome well above

[12] For example, a sample whose test score varies only within 5% would be very insufficient since most people are of the same ability and even the test prediction itself may not be that accurate.

that of the fixed sequence performance for the UNIX test, but only at about the same level for the French test. The performance of IRT is distinctively lower for IRT over both test, but it does perform as well as POKS for the UNIX test after the tenth question.

Given that the POKS approach is computationally much simpler than the IRT approach, which relies on maximum-likelihood estimation techniques that can prove relatively complex for IRT (see Baker, 1992), these results are encouraging. However, beyond these findings, a number of issues remain. We discuss some of the major ones below.

Including concepts in POKS structures

Estimating the mastery of concepts is essential to provide the fine grained knowledge assessment that many learning environments often require. The POKS approach must provide some means of assessing concepts.

Like other graph modeling techniques, we can include concepts and skills within POKS structures. However, the manner in which items should be linked to concepts, and how concepts should be linked among themselves is an open question. Numerous schemes have been proposed for linking items to concepts, such as leaky AND/OR gates (Martin & Vanlehn, 1995; Conati et al., 2002), dynamic Bayesian networks (Mayo & Mitrovic, 2001), weighted means (Millán & Pérez-de-la-Cruz, 2002), and BN organized in a number of topologies as can be found in Vomlel (2004).

For POKS structures, one possibility is to treat concepts in exactly the same way as item nodes, and link them with *surmise* relations. For example, the mastery of a concept by some examinee can be independently assessed (such as in Vomlel, 2004), and the induction of the structure can proceed in much the same process as that described in the *POKS network induction* section. Preliminary exploration of this simple scheme seems to suggest that the results are not very positive and further investigation is necessary to confirm and explain such findings.

Another recently explored approach used the data from Vomlel (2004) in which concept mastery is derived from item responses using a perceptron (a single layered neural network, Desmarais et al. (2005)). The POKS updating algorithm serves to determine the probability of mastery of each item as new items are answered, and the new probabilities are, in turn, fed to the perceptron to determine concept mastery. The approach is compared to Vomlel's own predictions. The results show that although POKS is better than the BN constructed by Vomlel for predicting the examinee's success to individual question items, it is less accurate for predicting concept mastery. These results suggest that a BN can effectively take advantage of intra-concept relationships for predicting concept mastery, but that intra-item (item-to-item) relationships are more effective for predicting the outcome of individual item success. However, note that such approaches are somewhat impractical because they imply training with independently assessed concept mastery: in practice, one rarely has the luxury of knowing true examinee concept mastery.

Finally, the simplest way of including concepts into POKS is to use the traditional breakdown that teachers do. Subject matters are divided into a hierarchy of more and more specific topics. Items are the leaves of this hierarchy and a weighted mean is used to estimate mastery of the next level down. Note that the structure does not need to be a pure hierarchy and that a single item can belong to many concept/skill nodes. Exams are often structured this way. The accuracy of this method is directly linked to the accuracy of the leave nodes mastery estimates (test items) and the validity of the weighted means. This approach may not have the ability to model misconceptions and non linear combinations of items and concepts, but it has the quality of being universally used in schools and understood by everyone.

Furthermore, that approach avoids problem of estimating concepts independently for constructing a graph model and for calibrating conditional probabilities. In fact, in our view, that problem plagues graph models in general. Modeling with hidden nodes is very difficult to envision by non statisticians.

Automated learning constraint

POKS is an algorithmic learning/calibration approach. Structures such as Figure 3's are built automatically. This approach shares the same advantages as IRT in that respect. However, as a graphical modeling approach, it also has the expressiveness of these models, namely that items can be aggregated into concepts and further aggregated into hierarchies of concepts. Techniques such as those of VanLehn, Niu, Siler, and Gertner (1998) can be used for modeling the “concept” part of the network that stands above the test items. Alternatively, a concept can be defined as a function of the probability of mastery of a set of items. For example, it can be a weighted average of the probability of set of items which composes a concept (Millán & Pérez-de-la-Cruz, 2002).

However, for the purpose of this study, the network is defined solely over the test items and no concepts nodes are included. Imposing this requirement relieves us from any knowledge engineering effort to construct the network and thus makes the approach more comparable to IRT than other Bayesian modeling approaches that would require a knowledge engineering step. The same data can be used for both approaches, thus allowing a comparison on an equal basis.

POKS's sensitivity to noise

One of the critical issue with the POKS approach is the problem of correcting errors due to noise. This is a direct consequence of pruning the bi-directionality of posterior updates, as explained in the *Evidence propagation directionality* section. This can result in nodes having no incoming, or no outgoing links. For example, a difficult item can often have many outgoing links, but no incoming links (i.e. no other item's success significantly increases its probability). It follows that this node's probability can only decrease according to POKS updating scheme. If, for some reason, an expert misses an easy item, these items with no incoming links (the more difficult ones in general) will see their probability decrease with no chance of being raised later on, until they are directly observed. Over test with a large density of relations between items, and tests with a higher chance for guessing, such noisy answers are bound to create these sticky errors. They will also tend to affect more significantly the performance at the end of the test when only a few items are not yet observed.

This weakness can explain the fact that, although the score for the French test shown in Figure 6 quickly reaches around 95%, it does not improve afterwards except at the very end, contrary to the UNIX test which displays a relatively steady improvement throughout the test. That failure to improve its performance as new evidence is provided is also apparent in the question predictive experiment reported in Figure 7, where POKS' performance drops below that of the fixed sequence performance after around one hundred items. Again, this is consistent with the fact that sticky errors will accumulate and more significantly affect the performance at the end of a test.

Why that phenomenom is present only in the French test is unknown and we can only speculate at this point. It can be related to the amount of "noise" in the data, such as lucky guesses. It can also be that the French test items are not as well structured as the UNIX ones, resulting in more errors over the nodes with only incoming or outgoing links. Whatever the reason is, the problem is not insurmountable, but it does involve developing some means to avoid the accumulation of noise over items that are either very difficult or very easy.

CONCLUSION

POKS offers a fully algorithmic means of building the model and updating item probabilities among themselves without requiring any knowledge engineering step. Indeed, the specific POKS approach uses the same data as the IRT-2PL approach to provide similar accuracy. It shows that a graphical modeling modeling approach such as POKS can be induced from a small amount of test data to perform relatively accurate examinee classification. This is an important feature from a practical perspective since the technique can benefit a large number of application contexts.

The graphical modeling approaches such as POKS or as Bayesian networks are still in their infancy compared to the IRT techniques developed since the 1960's, and their potential benefit remains relatively unexplored. However, applications of CAT techniques to tutoring systems and to different learning environments are emerging (see, for example, Millán, Garcia-Herve, Rueda, & de-la Cruz, 2003; Gonçalves, Aluisio, de Oliveira, & Oliveira, 2004; Conejo, Guzman, Millán, Trella, Pérez-de-la Cruz, & Rios, 2004). The availability of simple and automated techniques that are both effective and efficient, relying on little data and allowing seamless updates of test content, will be critical to their success in commercial applications.

Acknowledgements

This work has been supported by the National Research Council of Canada. The authors are grateful to the anonymous reviewers who provided generous and useful comments over previous versions of this paper.

REFERENCES

Baker, F. B. (1992). *Item response theory parameter estimation techniques*. New York, NY: Marcel Dekker Inc.

Birnbaum, A. (1968). Some latent trait models and their use in infering an examinee's ability. In F. Lord, & M. Novick (Eds.), *Statistical theories of mental test scores* (pp. 397–472). Reading, MA: Addison-Wesley.

Brusilovsky, P., Eklund, J., & Schwarz, E. (1998). Web-based education for all: A tool for developing adaptive courseware. *Proceedings of Seventh International World Wide Web Conference* (pp. 291–300). Brisbane, Australia.

Cheng, J., Greiner, R., Kelly, J., Bell, D., & Liu, W. (2002). Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence*, *137*(1–2), 43–90.

Collins, J. A. (1996). Adaptive testing with granularity. Master's thesis, University of Saskatchewan, Department of Computer Science.

Collins, J. A., Greer, J. E., & Huang, S. X. (1996). Adaptive assessment using granularity hierarchies and bayesian nets. *Intelligent Tutoring Systems* (pp. 569–577). Montreal, Canada.

Conati, C., Gertner, A., & VanLehn, K. (2002). Using bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, *12*(4), 371–417.

Conejo, R., Guzman, E., Millán, E., Trella, M., Pérez-de-la Cruz, J. L., & Rios, A. (2004). SIETTE: A web-based tool for adaptive teaching. *International Journal of Artificial Intelligence in Education*, *14*, 29–61.

Desmarais, M. C., Maluf, A., & Liu, J. (1995). User-expertise modeling with empirically derived probabilistic implication networks. *User Modeling and User-Adapted Interaction*, *5*(3-4), 283–315.

Desmarais, M. C., Meshkinfam, P., & Gagnon, M. (2005). *Bayesian modeling with strong vs. weak assumptions in the domain of skills assessment* (Technical report). Montreal, Canada: Ecole Polytechnique de Montreal.

Eggen, T. J. (1998). Item selection in adaptive testing with the sequential probability ratio test. *Applied Psychological Measurement*, *23*, 249–261.

Falmagne, J.-C., Koppen, M., Villano, M., Doignon, J.-P., & Johannesen, L. (1990). Introduction to knowledge spaces: How to build test and search them. *Psychological Review*, *97*, 201–224.

Giarratano, J., & Riley, G. (1998). *Expert systems: Principles and programming (3rd edition)*. Boston, MA: PWS-KENT Publishing.

Gonçalves, J. P., Aluisio, S. M., de Oliveira, L. H., & Oliveira, O. N. J. (2004). A learning environment for english for academic purposes based on adaptive tests and task-based systems. *Lecture Notes in Computer Science*, *3220*, 1–11.

Heckerman, D. (1995). *A tutorial on learning with bayesian networks* (Technical Report MSR-TR-95-06). Redmond, WA: Microsoft Research (MSR).

Jameson, A. (1995). Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction*, *5*(3-4), 193–251.

Jensen, F., Kjærul, U. B., Lang, M., & Madsen, A. L. (2002). Hugin - the tool for bayesian networks and influence diagrams. In J. A. Gámez, & A. Salmeron (Eds.), *Proceedings of the First European Workshop on Probabilistic Graphical Models, PGM 2002* (pp. 211–221).

Kambouri, M., Koppen, M., Villano, M., & Falmagne, J.-C. (1994). Knowledge assessment: tapping human expertise by the query routine. *International Journal of Human-Computer Studies*, *40*(1), 119–151.

Lewis, C., & Sheehan, K. (1990). Using bayesian decision theory to design a computerized mastery test. *Applied Psychological Measurement*, *14*(2), 367–386.

Liu, J., & Desmarais, M. (1997). A method of learning implication networks from empirical data: Algorithm and Monte-Carlo simulation-based validation. *IEEE Transactions on Knowledge and Data Engineering*, *9*(6), 990–1004.

Lord, F. M., & Novick, M. R. (Eds.). (1968). *Statistical theories of mental test scores*. Reading, MA: Addison-Weslay.

Martin, J., & Vanlehn, K. (1995). Student assessment using bayesian nets. *International Journal of Human-Computer Studies*, *42*(6), 575–591.

Mayo, M., & Mitrovic, A. (2001). Optimising ITS behaviour with bayesian networks and decision theory. *International Journal of Artificial Intelligence in Education*, *12*, 124–153.

McDonald, R. P. (1997). Normal-ogive multidimensional model. In W. J. van der Linden, & R. K. Hambleton (Eds.), *Handbook of modern item response theory* (pp. 257–286). Springer-Verlag, New York.

Millán, E., Garcia-Herve, E., Rueda, A., & de-la Cruz, J. P. (2003). Adaptation and generation in a web-based tutor for linear programming. *Lecture Notes in Computer Science*, *2722*, 124–127.

Millán, E., & Pérez-de-la-Cruz, J. L. (2002). A bayesian diagnostic algorithm for student modeling and its evaluation. *User Modeling and User-Adapted Interaction*, *12*(2–3), 281–330,.

Millán, E., Trella, M., Pérez-de-la-Cruz, J.-L., & Conejo, R. (2000). Using bayesian networks in computerized adaptive tests. In M. Ortega, & J. Bravo (Eds.), *Computers and education in the 21st century* (pp. 217–228). Kluwer.

Mislevy, R., & Chang, H. (2000). Does adaptive testing violate local independence? *Psychometrika*, *65*, 149–156.

Mislevy, R. J., & Gitomer, D. (1995). The role of probability-based inference in an intelligent tutoring system. *User Modeling and User-Adapted Interaction*, *42*(5), 253–282.

Neapolitan, R. E. (1998). *Probabilistic reasoning in expert systems: Theory and algorithms*. New York, NY: John Wiley & Sons, Inc.

Reckase, M. D. (1997). A linear logistic multidimensional model. In W. J. van der Linden, & R. K. Hambleton (Eds.), *Handbook of modern item response theory* (pp. 271–286). New York: Springer-Verlag.

Reye, J. (2004). Student modelling based on belief networks. *International Journal of Artificial Intelligence in Education*, *14*, 63–96.

Rudner, L. M. (2002). An examination of decision-theory adaptive testing procedures. *Proceedings of American Educational Research Association* (pp. 437–446). New Orleans.

van der Linden, W. J., & Hambleton, R. K. (Eds.). (1997). *Handbook of modern item response theory*. Springer-Verlag.

VanLehn, K., Lynch, C., Schulze, K., Shapiro, J., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., & Wintersgill, M. (2005). The andes physics tutoring system: Five years of evaluation (pp. 678–685).

VanLehn, K., & Martin, J. (1997). Evaluation of an assessment system based on bayesian student modeling. *International Journal of Artificial Intelligence in Education*, *8*, 179–221.

VanLehn, K., & Niu, Z. (2001). Bayesian student modeling, user interfaces and feedback: A sensitivity analysis. *International Journal of Artificial Intelligence in Education*, *12*, 154–184.

VanLehn, K., Niu, Z., Siler, S., & Gertner, A. S. (1998). Student modeling from conversational test data: A bayesian approach without priors. *ITS'98: Proceedings of the 4th International Conference on Intelligent Tutoring Systems* (pp. 434–443). London, UK: Springer-Verlag.

Venables, W. N., Smith, D. M., & the R Development Core Team (2004). *An introduction to R, notes on R: A programming environment for data analysis and graphics* (Technical report). R Project.

Vomlel, J. (2004). Bayesian networks in educational testing. *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems*, *12*(Supplementary Issue 1), 83–100.

Vos, H. J. (1999). Using bayesian decision theory to design a computerized mastery test. *Journal of Educational and Behavioral Statistics*, *24*(3), 271–292.

Zapata-Rivera, J.-D., & Greer, J. E. (2004). Interacting with bayesian student models. *International Journal of Artificial Intelligence in Education*, *14*(2), 127–163.