Knowledge Spaces and Learning Spaces[1]

Jean-Paul Doignon[2] Jean-Claude Falmagne[3]

Abstract

How to design automated procedures which (i) accurately assess the knowledge of a student, and (ii) efficiently provide advices for further study? To produce well-founded answers, Knowledge Space Theory relies on a combinatorial viewpoint on the assessment of knowledge, and thus departs from common, numerical evaluation. Its assessment procedures fundamentally differ from other current ones (such as those of S.A.T. and A.C.T.). They are adaptative (taking into account the possible correctness of previous answers from the student) and they produce an outcome which is far more informative than a crude numerical mark. This chapter recapitulates the main concepts underlying Knowledge Space Theory and its special case, Learning Space Theory. We begin by describing the combinatorial core of the theory, in the form of two basic axioms and the main ensuing results (most of which we give without proofs). In practical applications, learning spaces are huge combinatorial structures which may be difficult to manage. We outline methods providing efficient and comprehensive summaries of such large structures. We then describe the probabilistic part of the theory, especially the Markovian type processes which are instrumental in uncovering the knowledge states of individuals. In the guise of the ALEKS system, which includes a teaching component, these methods have been used by millions of students in schools and colleges, and by home schooled students. We summarize some of the results of these applications.

MSC-class: 91E45

1 Origin and Motivation

Knowledge Space Theory (abbreviated as KST) originated with a paper by Doignon and Falmagne (1985). This work was motivated by the shortcomings of the psychometric approach to the assessment of competence. The psychometric models are based on the notion that competence can be measured, which the two authors thought was at least debatable. Moreover, a typical application of a psychometric model in the form of a standardized test results in placing an individual in one of a few dozen ordered categories, which is far too coarse a classification to be useful. In the case of the S.A.T.[4], for example, the result

[1] A different version is to appear in *The New Handbook of Mathematical Psychology*, Cambridge University Press. The authors thank Laurent Fourny and Keno Merckx for their careful reading of a preliminary draft.

[2] Université Libre de Bruxelles, Département de Mathématique c.p. 216, B-1050 Bruxelles, Belgium. doignon@ulb.ac.be

[3] Department of Cognitive Sciences, University of California, Irvine, CA 92617, USA, jcf@uci.edu

[4] A test for college admission and placement in the U.S. The acronym S.A.T used to mean "Scholastic Aptitude Test". A few years ago the meaning of S.A.T. was changed into "Scholastic Assessment Test". Today this acronym stands alone, without any associated meaning.

of the test is a number between 200 and 800 with only multiples of 10 being possible scores.

In the cited paper, Doignon and Falmagne proposed a fundamentally different theory. The paper was followed by many others, written by them and other researchers (see the Bibliographical Notes in Section 13).

The basic idea is that an assessment in a scholarly subject should uncover the individual's 'knowledge state', that is, the exact set of concepts mastered by the individual. Here, 'concept' means a type of problem that the individual has learned to master, such as, in Beginning Algebra:

solving a quadratic equation with integer coefficients;

or, in Basic Chemistry

balance a chemical equation using the smallest
whole number stoichiometric coefficients[5].

In KST, a problem type is referred to as an 'item'. Note that this usage differs to that in psychometric, where an item is a particular problem, such as: *Solve the quadratic equation* $x^2 - x - 12 = 0$. In our case, the examples of an item are called *instances*[6].

The items or problem types form a possibly quite large set, which we call the 'domain' of the body of knowledge. A knowledge state is a subset of the domain, but not any subset is a state: the knowledge states form a particular collection of subsets, which is called the 'knowledge structure' or more specifically (when certain requirements are satisfied) the 'knowledge space' or the 'learning space'. The collection of states captures the whole structure of the domain. As in Beginning Algebra, which will be our lead example in this chapter, the domain may contain as many as 650 items, and the learning space may contain many millions states, in sharp contrast with the few dozen scoring categories of a psychometric test. Despite this large number of possible knowledge states, an efficient assessment is feasible in the course of 25–35 questions.

In Sections 2 to 8, we review the fundamental combinatorial concepts and the main axiomatizations. We introduce the important special case of KST, Learning Space Theory (LST). As the collection of all the feasible, realistic knowledge states may be very large, it is essential to find efficient summaries. We describe two such summaries in Sections 4 and 7. The theory discussed in this chapter has been extensively applied in the schools and universities. The success of the applications is due in large part to a feature of the knowledge state produced by the assessment: the state is predictive of what a student is ready to learn. The reason lies in a formal result, the 'Fringe Theorem' (see Section 6). In some situations, it is important to focus on a part of a knowledge structure. We call the relevant concept a 'projection' of a knowledge structure on a subset of the items. This is the subject of Section 8.

[5] For example: $\mathrm{Fe_2O_3}(s) \rightarrow \mathrm{Fe}(s)+\mathrm{O_2}(g)$, which is not balanced. The correct response is: $2\mathrm{Fe_2O_3}(s) \rightarrow 4\mathrm{Fe}(s) + 3\mathrm{O_2}(g)$.

[6] So, the instances of knowledge space theory are the items of psychometrics.

A next section is devoted to the description of the Markovian type stochastic assessment procedure (Section 10). It relies on the notion of a probabilistic knowledge structure, introduced in Section 9. In Section 11, we give an outline of our methods for building the fundamental structure of states for a particular scholarly domain, such as Beginning Algebra, Pre-Calculus, or Statistics. Such constructions are enormously demanding and time consuming. They rely not only on dedicated mathematical algorithms, but also on huge bases of assessment data. The most extensive applications of KST are in the form of the web based system called ALEKS[7], which includes a teaching component. Millions of students have used the system, either at home, or in schools and universities. Section 12 reports some results of these applications.

This chapter summarizes key concepts and results from two books. One is the monograph of Falmagne and Doignon (2011)[8]. The other book is the edited volume of Falmagne et al. (2013), which contains recent data on the applications of the theory and also some new theoretical results. A few additional results appear here in Section 7 and 11.

2 Knowledge Structures and Learning Spaces

We formalize the cognitive structure of a scholarly subject as a collection $\mathcal{K}$ of subsets of a basic set Q of items. In the case of Beginning Algebra, the items forming Q are the types of problems a student must master to be fully conversant in the subject. We suppose that the collection $\mathcal{K}$ contains at least two subsets: the empty set, which is that of a student knowing nothing at all in the subject, and the full set Q of problems. The next definition cast these basic notions in set-theoretic terms. We illustrate the definition by a few examples.

Definition 1. A *knowledge structure* is a pair $(Q, \mathcal{K})$ consisting of a nonempty set Q and a collection $\mathcal{K}$ of subsets of Q; we assume $\varnothing \in \mathcal{K}$ and $Q \in \mathcal{K}$. The set Q is called the *domain* of the knowledge structure $(Q, \mathcal{K})$. The elements of Q are the *items*, and the elements of $\mathcal{K}$ are the *knowledge states*, or just the *states*. A knowledge structure $(Q, \mathcal{K})$ is *finite* when its domain Q is a finite set.

For any item q in Q, we write $\mathcal{K}_q$ for $\{K \in \mathcal{K} \mid q \in K\}$, the subcollection of $\mathcal{K}$ consisting of all the states containing q. A knowledge structure $(Q, \mathcal{K})$ is *discriminative* when for any two items q and r in the domain, we have $\mathcal{K}_q = \mathcal{K}_r$ only if $q = r$.

We often abbreviate $(Q, \mathcal{K})$ into $\mathcal{K}$ (with no loss of information because $Q = \cup\mathcal{K}$).

[7] ALEKS is an acronym for "**A**ssessment and **LE**arming in **K**nowledge **S**paces".

[8] This is a much expanded reedition of Doignon and Falmagne (1999).

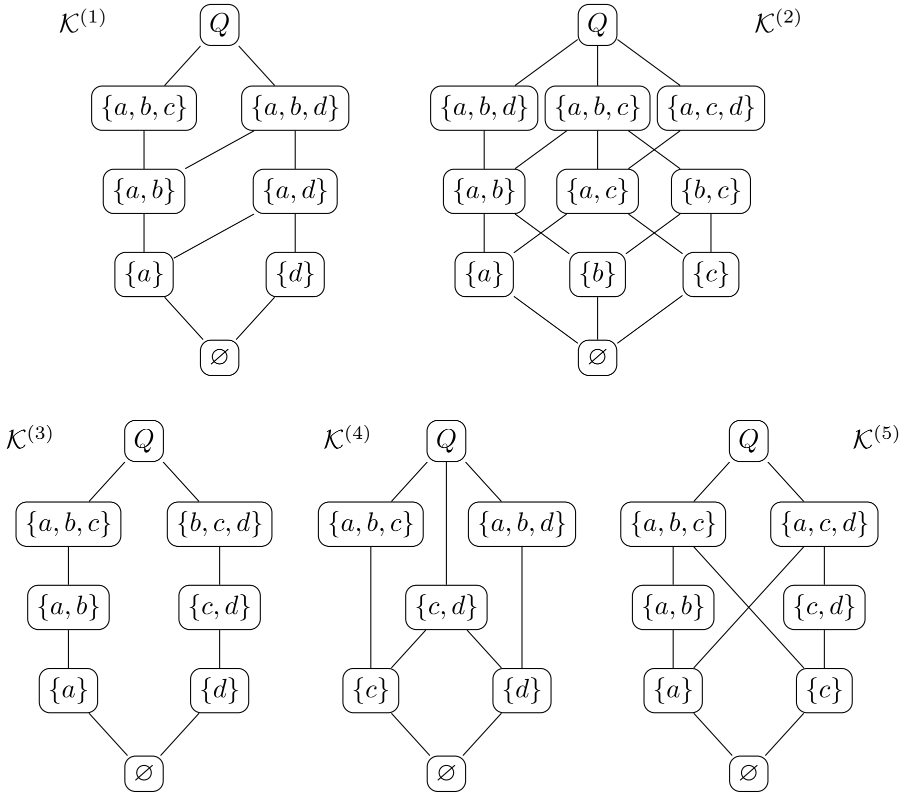

Figure 1: The five examples of knowledge structures in Example 2.

Example 2. Here are five examples of knowledge structures all on the same domain $Q = \{a,b,c,d\}$:

$$\begin{aligned}
\mathcal{K}^{(1)} &= \{\varnothing, \{a\}, \{d\}, \{a,b\}, \{a,d\}, \{a,b,c\}, \{a,b,d\}, Q\},\\
\mathcal{K}^{(2)} &= \{\varnothing, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,d\}, \{a,b,c\}, \{a,c,d\}, Q\},\\
\mathcal{K}^{(3)} &= \{\varnothing, \{a\}, \{d\}, \{a,b\}, \{c,d\}, \{a,b,c\}, \{b,c,d\}, Q\},\\
\mathcal{K}^{(4)} &= \{\varnothing, \{c\}, \{d\}, \{c,d\}, \{a,b,c\}, \{a,b,d\}, Q\},\\
\mathcal{K}^{(5)} &= \{\varnothing, \{a\}, \{c\}, \{a,b\}, \{c,d\}, \{a,b,c\}, \{a,c,d\}, Q\}.
\end{aligned}$$

The five knowledge structures are finite, and all but $\mathcal{K}^{(4)}$ are discriminative: we have

$$\mathcal{K}^{(4)}_a = \mathcal{K}^{(4)}_b = \{\{a,b,c\}, \{a,b,d\}, Q\} \qquad \text{with} \quad a \neq b. \tag{1}$$

In the graphs of these knowledge structures displayed in Figure 1, the ascending lines show the covering relation of the states; that is, we have an ascending line from the point representing the state K to the one representing the state L

exactly when K is *covered* by L, that is when $K \subset L$ and moreover there is no state A in $\mathcal{K}$ such that $K \subset A \subset L$.

We call a representation of a knowledge structure as exemplified in Figure 1 a *covering diagram* of the structure. Note in passing two extreme cases of knowledge structures on a given domain Q. One is $(Q, 2^Q)$, where 2^Q denotes the *power set* of Q, that is the collection of all the subsets of Q. The other one is $(Q, \{\varnothing, Q\})$, in which the knowledge structure contains only the two required states $\varnothing$ and Q. These two examples are trivial and uninteresting because they entail a complete lack of organization in the body of information covered by the items in Q.

Two requirements on a knowledge structure make good pedagogical sense. One is that there should be no gaps in the organization of the material: the student should be able to master the items one by one. Also, there should be some consistency in the items: an advanced student should have less trouble learning a new item than another, less competent student has. The two conditions incorporated in the next definition formalize the two ideas.

Definition 3. A *learning space* $(Q, \mathcal{K})$ is a knowledge structure which satisfies the two following conditions:

[L1] LEARNING SMOOTHNESS. For any two states K, L with $K \subset L$, there exists a finite chain of states

$$K = K_0 \subset K_1 \subset \cdots \subset K_p = L \tag{2}$$

such that $|K_i \setminus K_{i-1}| = 1$ for $1 \leq i \leq p$ (thus we have $|L \setminus K| = p$).

In words: *If the learner is in some state K included in some state L, then the learner can reach state L by mastering items one by one.*

[L2] LEARNING CONSISTENCY. For any two states K, L with $K \subset L$, if q is an item such that $K \cup \{q\}$ is a state, then $L \cup \{q\}$ is also a state.

In words: *Knowing more does not prevent learning something new.*

Notice that any learning space is finite. Indeed Condition [L1] applied to the two states $\varnothing$ and Q yields a finite chain of states from $\varnothing$ to Q. In Example 2 (see also Figure 1), only the two structures $\mathcal{K}^{(1)}$ and $\mathcal{K}^{(2)}$ are learning spaces. The knowledge structure $\mathcal{K}^{(3)}$ satisfies Condition [L1] in Definition 3 but not Condition [L2] (take $K = \varnothing$, $L = \{a\}$ and $q = d$). As for $\mathcal{K}^{(4)}$, it satisfies [L2] but not [L1]. Finally, $\mathcal{K}^{(5)}$ does not satisfy either condition. A simpler way to check whether a covering diagram as in Figure 1 represents a learning space is provided in the next section, just after Theorem 8.

We give one more example of a learning space, which is realistic in that its ten items belong to the domain of the very large learning space of Beginning Algebra used in the `ALEKS` system. A remarkable feature of a learning space is that any subset of the items of a learning space also defines a learning space (see below Definition 31 and Theorem 33). Thus we have a learning space on these ten items. This example will be used repeatedly later on, and in particular to illustrate the assessment mechanism, that is, the questioning algorithm uncovering the knowledge state of a student.

Table 1: The items of the ten-item example of Figure 2.

a. Quotients of expressions involving exponents	**b.** Multiplying two binomials
c. Plotting a point in the coordinate plane using a virtual pencil on a Cartesian graph	**d.** Writing the equation of a line given the slope and a point on the line
e. Solving a word problem using a system of linear equations (advanced problem)	**f.** Graphing a line given its equation
g. Multiplication of a decimal by a whole number	**h.** Integer addition (introductory problem)
i. Equivalent fractions: fill the blank in the equation $\frac{a}{b} = \frac{\square}{c}$, where a, b and c are whole numbers	**j.** Graphing integer functions

Example 4. Ten items in Beginning Algebra. Table 1 lists ten items. Remember that items are types of problems, and not particular cases (which are called instances). Here is an instance of item **d**: *A line passes through the point* $(x, y) = (-3, 2)$ *and has a slope of* 6. *Write an equation for this line.* Figure 2 shows 34 knowledge states in $Q = \{a, b, \ldots, j\}$ which together form a learning space.

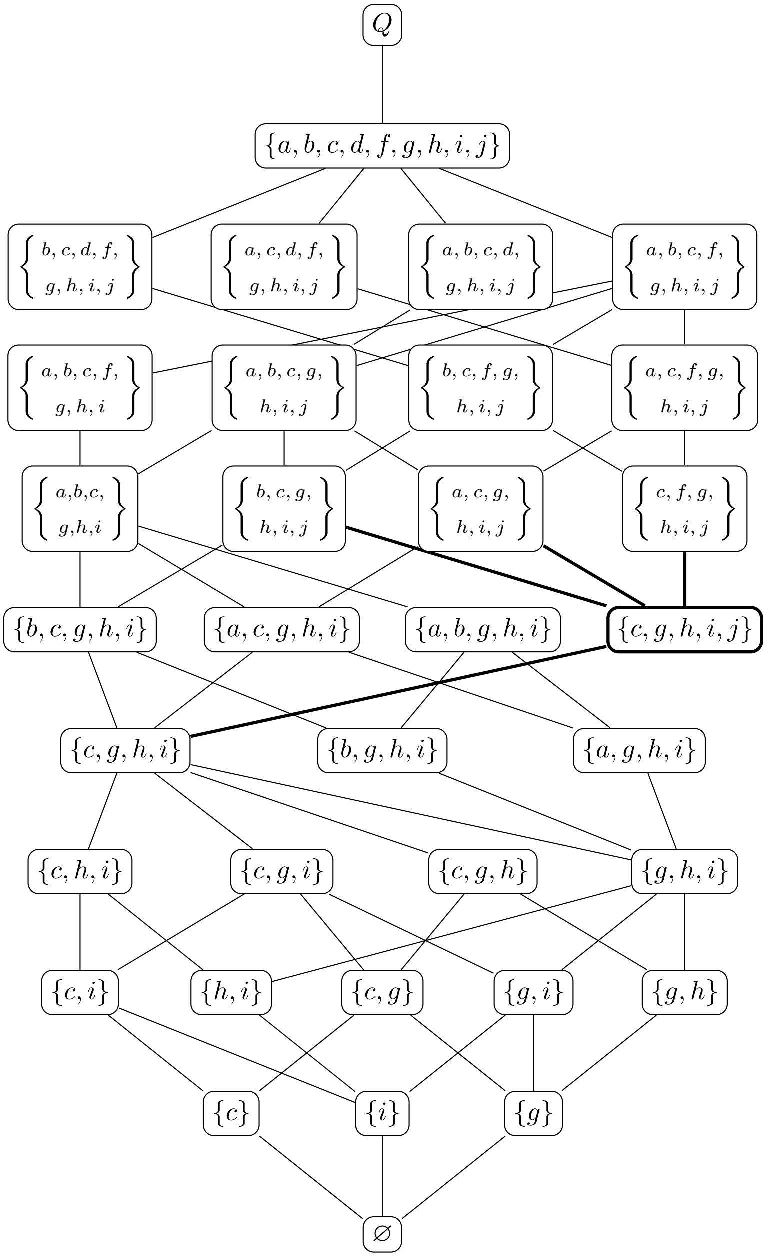
Q
$\{a, b, c, d, f, g, h, i, j\}$
$\left\{\begin{matrix} b, c, d, f, \\ g, h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, c, d, f, \\ g, h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, b, c, d, \\ g, h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, b, c, f, \\ g, h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, b, c, f, \\ g, h, i \end{matrix}\right\}$
$\left\{\begin{matrix} a, b, c, g, \\ h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} b, c, f, g, \\ h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, c, f, g, \\ h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a,b,c, \\ g,h,i \end{matrix}\right\}$
$\left\{\begin{matrix} b, c, g, \\ h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} a, c, g, \\ h, i, j \end{matrix}\right\}$
$\left\{\begin{matrix} c, f, g, \\ h, i, j \end{matrix}\right\}$
$\{b, c, g, h, i\}$
$\{a, c, g, h, i\}$
$\{a, b, g, h, i\}$
$\{c, g, h, i, j\}$
$\{c, g, h, i\}$
$\{b, g, h, i\}$
$\{a, g, h, i\}$
$\{c, h, i\}$
$\{c, g, i\}$
$\{c, g, h\}$
$\{g, h, i\}$
$\{c, i\}$
$\{h, i\}$
$\{c, g\}$
$\{g, i\}$
$\{g, h\}$
$\{c\}$
$\{i\}$
$\{g\}$
$\varnothing$

Figure 2: The covering diagram of the ten-item learning space $\mathcal{L}$ of Example 4. The meaning of the four bold lines joining the state $\{c, g, h, i, j\}$ to four other states is explained in Section 6.

3 Knowledge Spaces and Wellgradedness

The name "learning space" specifically refers to Conditions [L1] and [L2] of Definition 3. As mentioned earlier, the two conditions have an interesting, pedagogical interpretation. Other characterizations of the same combinatorial concept focus on some other key concepts, which we review in the present section. The *symmetric difference* between two sets K and L is defined by $K \triangle L = (K \setminus L) \cup (L \setminus K)$.

Definition 5. A *knowledge space* $\mathcal{K}$ is a knowledge structure which is closed under union, or *$\cup$-closed*, that is, $\cup\mathcal{C} \in \mathcal{K}$ for any subcollection $\mathcal{C}$ of $\mathcal{K}$. The knowledge structure $\mathcal{K}$ is *well-graded* if for any two states K and L in $\mathcal{K}$, there exists a natural number h such that $|K \triangle L| = h$ and a finite sequence of states $K = K_0, K_1, \ldots, K_h = L$ such that $|K_{i-1} \triangle K_i| = 1$ for $1 \leq i \leq h$. The knowledge structure $\mathcal{K}$ is *accessible* or *downgradable*[9] if for any nonempty state K in $\mathcal{K}$, there is some item q in K such that $K \setminus q \in \mathcal{K}$. A downgradable, finite knowledge space is called an *antimatroid*[10].

The closure under union is a critical property because it enables a (sometime highly efficient) summary of a knowledge space by a minimal subcollection of its states. The subcollection is called the 'base' of the knowledge space and is one of the topics of our next section. Note that a well-graded knowledge structure $(Q, \mathcal{K})$ is necessarily accessible (given the state K, take $L = \varnothing$ in Definition 5). However, an accessible knowledge structure is not necessarily finite nor well-graded.

Example 6. Take as items all the natural numbers, thus the domain is $\mathbb{N}$. As states, take the empty set plus all the subsets of $\mathbb{N}$ whose complement is finite. We denote by $\mathcal{G}$ the collection of states:

$$\mathcal{G} \;=\; \{\varnothing\} \cup \{K \in 2^{\mathbb{N}} \;\big|\; |\mathbb{N} \setminus K| < +\infty\}. \tag{3}$$

The resulting structure $(\mathbb{N}, \mathcal{G})$ is accessible. It is infinite, and not well-graded (consider for instance the two states $\varnothing$ and $\mathbb{N}$).

Theorem 7. *For any knowledge structure $(Q, \mathcal{K})$, the following three statements are equivalent.*

(i) $(Q, \mathcal{K})$ *is a learning space.*

(ii) $(Q, \mathcal{K})$ *is an antimatroid.*

(iii) $(Q, \mathcal{K})$ *is a well-graded knowledge space.*

Cosyn and Uzun (2009) proved the equivalence of Conditions (i) and (ii) in Theorem 7, while Korte et al. (1991) established still another characterization of antimatroids (or learning spaces): Theorem 8 below is Lemma 1.2 of their Chapter 3. We provide a combined proof of Theorems 7 and 8 below.

[9] See Doble et al. (2001) for the latter term.

[10] See Korte et al. (1991) for this use of the word. For other authors an antimatroid is closed under intersection rather than under union (and "upgradable" rather than downgradable), see for example Edelman and Jamison (1985).

Theorem 8. *A knowledge structure $(Q, \mathcal{K})$ is a learning space if and only if its collection $\mathcal{K}$ of states satisfies the following three conditions:*

(a) *Q is finite;*

(b) *$\mathcal{K}$ is downgradable, that is: any nonempty state K contains some item q such that $K \setminus \{q\} \in \mathcal{K}$;*

(c) *for any state K and any items q, r, if $K \cup \{q\}$, $K \cup \{r\} \in \mathcal{K}$, then $K \cup \{q, r\} \in \mathcal{K}$.*

Theorem 8 makes it easy to check whether a (finite) covering diagram such as that pictured in Figure 1 represents a learning space. Assume that the points representing two states are at the same level (or height) if and only if they have the same number of items; then it suffices to check that: (i) any ascending line connects points at two successive levels; (ii) any point representing a nonempty state is the end of at least one ascending line; (iii) if two ascending lines start from the same point, then their endpoints are the origins of ascending lines having the same endpoint.

Proofs of Theorems 7 and 8. For a given knowledge structure $(Q, \mathcal{K})$, we show that

$$\text{(i)} \Rightarrow \text{(ii)} \Rightarrow \text{(iii)} \Rightarrow \big(\text{(a), (b) and (c)}\big) \Rightarrow \text{(i)}.$$

First notice that any of the three conditions (i), (ii), (iii) entails the finiteness of Q, that is, Condition (a).

(i) $\Rightarrow$ (ii). Let $(Q, \mathcal{K})$ be a learning space as in Definition 3. We prove that $(Q, \mathcal{K})$ is an antimatroid as in Definition 5, in other words that $(Q, \mathcal{K})$ is a finite knowledge space which is moreover downgradable. Suppose that K, L are states. We first apply Learning Smoothness to $\varnothing$ and L and derive a sequence $L_0 = \varnothing$, L_1, …, $L_\ell = L$ of states such that $|L_i \setminus L_{i-1}| = 1$ for $1 \leq i \leq \ell$. Then applying Learning Consistency to the states $\varnothing$ and K and the item forming L_1, we derive $K \cup L_1 \in \mathcal{K}$. Next, we apply Learning Consistency to the states L_1 and $K \cup L_1$ and the item forming $L_2 \setminus L_1$ to derive $K \cup L_2 \in \mathcal{K}$. The general step, for $i = 1$, 2, …, ℓ, applies Learning Consistency to L_{i-1} and $K \cup L_{i-1}$ and the item forming $L_i \setminus L_{i-1}$ to derive $K \cup L_i \in \mathcal{K}$. At the last step ($i = \ell$), we get $K \cup L \in \mathcal{K}$. On the other hand, downgradability of $\mathcal{K}$ at the state K is just a particular case of Learning Smoothness at the states $\varnothing$ and K.

(ii) $\Rightarrow$ (iii). If $(Q, \mathcal{K})$ is an antimatroid, then $\mathcal{K}$ is closed under union by definition. To prove the wellgradedness of $\mathcal{K}$ (Definition 5), we take two states K and L. By assumption, $K \cup L$ is also a state, and moreover by downgradability there exist a sequence of states $M_0 = \varnothing$, M_1, …, $M_h = K \cup L$ with $|M_i \setminus M_{i-1}| = 1$ for $1 \leq i \leq h$. Then $K \cup M_0$, $K \cup M_1$, …, $K \cup M_h$, after deletion of repetitions, becomes an increasing sequence K_0, K_1, …, K_k from K to $K \cup L$ with increments consisting of one item. We derive a similar sequence L_0, L_1, …, L_ℓ from L to $K \cup L$. Finally, K_0, K_1, …, $K_k = L_\ell$, $L_{\ell-1}$, …, L_0 is the required sequence from K to L (indeed, $k + \ell = |K \triangle L|$).

(iii) ⇒ ((a), (b) and (c)). Downgradability (b) is a direct consequence of well-gradedness. To prove (c), we only need to notice $K \cup \{q, r\} = (K \cup \{q\}) \cup (K \cup \{r\})$ and apply the assumed closure under union.

((a), (b) and (c)) ⇒ (i). To prove Learning Smoothness, we consider two states K and L such that $K \subset L$. By downgradability, there exist two sequences $K_0 = \varnothing$, K_1, …, $K_k = K$ and $L_0 = \varnothing$, L_1, …, $L_\ell = L$ of states such that $|K_i \setminus K_{i-1}| = 1$ for $1 \leq i \leq k$ and $|L_j \setminus L_{j-1}| = 1$ for $1 \leq j \leq \ell$. By repeated applications of (c), we derive $K_1 \cup L_1$, $K_2 \cup L_1$, …, $K_k \cup L_1 \in \mathcal{K}$, next $K_1 \cup L_2$, $K_2 \cup L_2$, …, $K_k \cup L_2 \in \mathcal{K}$, etc., and finally $K_1 \cup L_{l-1}$, $K_2 \cup L_{l-1}$, …, $K_k \cup L_{l-1} \in \mathcal{K}$. Thus $K \cup L_0 = K$, $K \cup L_1$, …, $K \cup L_{l-1}$, L are all in $\mathcal{K}$ (remember $K_k = K$, $L_l = L$ and $K \subset L$). After deletion of repetitions, we obtain the desired sequence from K to L.

To prove Learning Consistency, we again consider two states K and L with $K \subset L$ together with an item q such that $K \cup \{q\} \in \mathcal{K}$. In the previous paragraph, we proved the existence of a sequence $M_0 = K$, M_1, …, $M_h = L$ of states such that $|M_i \setminus M_{i-1}| = 1$ for $1 \leq i \leq h$. Applying (c) repeatedly, we obtain $M_1 \cup \{q\} \in \mathcal{K}$, $M_2 \cup \{q\} \in \mathcal{K}$, …, $M_h \cup \{q\} \in \mathcal{K}$, the last one being $L \cup \{q\} \in \mathcal{K}$ as desired. □

A simple case of a learning space arises when the collection of states is closed under both union and intersection.

Definition 9. A *quasi ordinal space* is a knowledge space closed under intersection. A *(partially) ordinal space* is a quasi ordinal space which is discriminative.

In Example 2, only the structure $\mathcal{K}^{(1)}$ is a quasi ordinal space, and it is even an ordinal space. The reason for the terminology in Definition 9 lies in Theorem 10 below, due to Birkhoff (1937). We recall that a *quasi order* on Q is a reflexive and transitive relation on Q. A *partial order* on Q is a quasi order on Q which is an *antisymmetric relation* (that is, for all q and r in Q, it holds that qRr and rRq implies $q = r$).

Theorem 10 (Birkhoff, 1937). *There exists a one-to-one correspondence between the collection of all quasi ordinal spaces $\mathcal{K}$ on a set Q and the collection of all quasi orders $\mathcal{Q}$ on Q. One such correspondence is specified by the two equivalences*[11]

$$\text{for all } q, r \text{ in } Q: \quad q\mathcal{Q}r \iff \mathcal{K}_q \supseteq \mathcal{K}_r; \tag{4}$$

$$\text{for all } K \subseteq Q: \quad K \in \mathcal{K} \iff (\forall (q, r) \in \mathcal{Q} : r \in K \Rightarrow q \in K). \tag{5}$$

Its restriction to discriminative spaces links partially ordinal spaces to partial orders.

Note in passing that the closure under intersection does not make good pedagogical sense. A variant of Theorem 10 for knowledge spaces appears below as Theorem 18; a variant for learning spaces follows from Theorem 19.

[11] We recall the formula $\mathcal{K}_q = \{K \in \mathcal{K} \mid q \in K\}$.

4 The Base and the Atoms

In practice, learning spaces tend to be very large, counting millions of states. For various purposes—for example, to store the structure in a computer's memory—such huge structures need to be summarized. One such summary is the 'base' of the structure, which we define below.

Definition 11. The *span* of a collection of sets $\mathcal{F}$ is the collection of sets $\mathcal{F}'$ containing exactly those sets that are unions of sets in $\mathcal{F}$. We say then that $\mathcal{F}$ spans $\mathcal{F}'$ and we write $\mathbb{S}(\mathcal{F}) = \mathcal{F}'$. So, $\mathbb{S}(\mathcal{F})$ is necessarily $\cup$-closed. A *base* of a $\cup$-closed collection $\mathcal{S}$ of sets is a minimal subcollection $\mathcal{B}$ of $\mathcal{S}$ spanning $\mathcal{S}$—where "minimal" refers to inclusion, that is, if $\mathbb{S}(\mathcal{H}) = \mathcal{S}$ for some $\mathcal{H} \subseteq \mathcal{B}$, then necessarily $\mathcal{B} \subseteq \mathcal{H}$.

Note that by a common convention, the empty set is the union of zero set in $\mathcal{B}$. Accordingly, the empty set never belongs to a base. It is easily shown that the base of a knowledge space is unique when it exists. Also, any finite knowledge space has a base (see Theorems 3.4.2 and 3.4.4 in Falmagne and Doignon, 2011). However, some $\cup$-closed collection of sets have no base; an example is the collection of all the open subsets of the set of real numbers. Any learning space has a base because it is finite and $\cup$-closed (cf. Theorem 7, (i) $\Leftrightarrow$ (iii)).

For example, the base of the learning space $\mathcal{K}^{(2)}$ displayed in Figure 1 is

$$\{\ \{a\},\ \{b\},\ \{c\},\ \{a, b, d\},\ \{a, c, d\}\ \}. \tag{6}$$

The economy is not great in this little example but may become spectacular in the case of the very large structures encountered in practice.

Another reason for the importance of the base stems from a pedagogical concept. The relevant question is: "*Given some item q, which minimal state, or states, must be mastered for q to be mastered?*". In more direct words: "*what are the minimal states containing a given item q?*". As one might guess, these minimal sets coincide with the elements of the base.

Definition 12. Let $\mathcal{K}$ be a knowledge space. For any item q, an *atom at q* is a minimal state of $\mathcal{K}$ containing q. A state K is called an *atom* if K is an atom at q for some item q. A knowledge space is *granular* if for any item q and any state K containing q, there is an atom at q which is included in K.

Clearly, any finite knowledge space is granular. On the other hand, a state K is an atom in a knowledge space $\mathcal{K}$ if and only if any subcollection of states $\mathcal{F}$ such that $K = \cup\mathcal{F}$ contains K (cf. Theorem 3.4.7 in Falmagne and Doignon, 2011). Note also that any granular knowledge space has a base (cf. Proposition 3.6.6 in Falmagne and Doignon, 2011).

Example 13. For the ten-item learning space pictured in Figure 2 (page 7), there are two atoms at item f, namely

$$\{c, f, g, h, i, j\}, \quad \{a, b, c, f, g, h, i\}. \tag{7}$$

Table 2: The atoms at all the items of the ten-item learning space from Example 4 (see Example 13).

Items	Atoms
a	$\{a,\ g,\ h,\ i\}$
b	$\{b, g, h, i\}$
c	$\{c\}$
d	$\{b, c, d, f, g, h, i, j\},\ \{a, c, d, f, g, h, i, j\},\ \{a, b, c, d, g, h, i, j\}$
e	Q
f	$\{c, f, g, h, i, j\},\ \{a, b, c, f, g, h, i\}$
g	$\{g\}$
h	$\{h, i\},\ \{g, h\}$
i	$\{i\}$
j	$\{c, g, h, i, j\}$

You can check from the figure that these two sets are indeed minimal states containing f and that they are the only ones with that property. Note that there is just one atom at the item b, which is $\{b,\ g,\ h,\ i\}$, while there are three atoms at d. Table 2 displays the full information on the atoms.

We conclude this section with the expected result (a proof is given in Falmagne and Doignon, 2011[12]).

Theorem 14. *Suppose that a knowledge space has a base. Then this base is exactly the collection of all the atoms.*

A simple algorithm, due to Dowling (1993) and grounded on the concept of an atom, constructs the base of a finite knowledge space given the states. In the same paper, she also describes a more elaborate algorithm for efficiently building the span of a collection of subsets of a finite set. Both algorithms are sketched in Falmagne and Doignon (2011, pages 49–50) (another algorithm for the second task, in another context, is due to Ganter; see Ganter and Reuter, 1991). The concept of an atom is closely related to that of the 'surmise system', which is the topic of our next section. We complete the present section with a characterization of learning spaces through their atoms (Koppen, 1998).

[12] The reader should refer to that monograph for most of the proofs omitted in this chapter.

Theorem 15. *For any finite knowledge structure $(Q, \mathcal{K})$, the following three statements are equivalent:*

(i) *$(Q, \mathcal{K})$ is a learning space;*

(i) *for any atom A at item q, the set $A \setminus \{q\}$ is a state;*

(i) *any atom is an atom at only one item.*

5 Surmise Systems

In a finite knowledge space, a student masters an item q only when his state includes some atom C at q. So, the collection of all the atoms at the various items may provide a new way to specify a knowledge space. We illustrate this idea by the following example of a knowledge space.

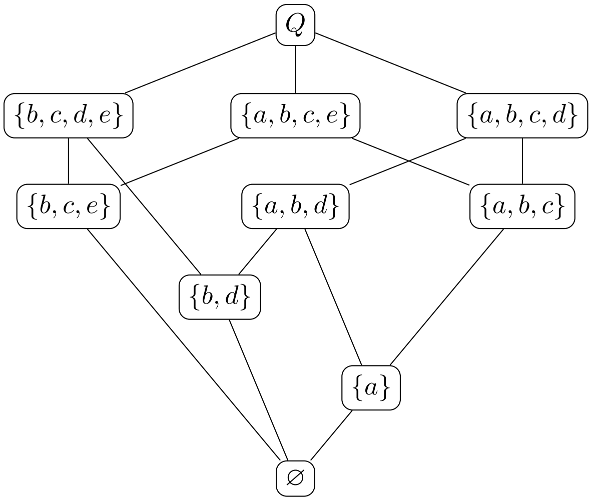

Figure 3: The covering diagram of the knowledge space in Example 16.

Example 16. Consider the knowledge space

$$\begin{aligned}\mathcal{H} = \big\{&\varnothing, \{a\}, \{b, d\}, \{a, b, c\}, \{a, b, d\}, \{b, c, e\},\\ &\{a, b, c, d\}, \{a, b, c, e\}, \{b, c, d, e\}, \{a, b, c, d, e\}\big\}\end{aligned} \tag{8}$$

on the domain $Q = \{a, b, c, d, e\}$. Figure 3 provides its covering diagram, while Table 3 lists its atoms. Table 3 links each of items a, d and e to a single atom of $\mathcal{H}$, and items b and c to three and two atoms, respectively. So, to master item b, one must first master either item d, or items a and c, or items c and e.

Example 16 illustrates the following definition.

Table 3: Items and their atoms in the knowledge space of Equation (8) (see also Figure 3).

Items	Atoms
a	$\{a\}$
b	$\{b,d\}$, $\{a,b,c\}$, $\{b,c,e\}$
c	$\{a,b,c\}$, $\{b,c,e\}$
d	$\{b,d\}$
e	$\{b,c,e\}$

Definition 17. Let Q be a nonempty set of items. A function $\sigma : Q \to 2^{2^Q}$ mapping each item q in Q to a nonempty collection $\sigma(q)$ of subsets of Q (so, $\sigma(q) \neq \varnothing$) is called an *attribution function* on the set Q. For each q in Q, any C in $\sigma(q)$ is called a *clause for* q (in σ). A *surmise function* σ on Q is an attribution function on Q which satisfies the three additional conditions, for all $q, q' \in Q$, and $C, C' \subseteq Q$:

(i) if $C \in \sigma(q)$, then $q \in C$;

(ii) if $q' \in C \in \sigma(q)$, then $C' \subseteq C$ for some $C' \in \sigma(q')$;

(iii) if $C, C' \in \sigma(q)$ and $C' \subseteq C$, then $C = C'$.

In such a case, the pair (Q, σ) is a *surmise system*. A surmise system (Q, σ) is *discriminative* if σ is injective (that is: whenever $\sigma(q) = \sigma(q')$ for some $q, q' \in Q$, then $q = q'$). Then the surmise function σ is also called *discriminative*.

It is easily shown that any attribution function σ on a set Q defines a knowledge space $(Q, \mathcal{K})$ via the equivalence

$$K \in \mathcal{K} \quad \Longleftrightarrow \quad \forall q \in K, \exists C \in \sigma(q) : C \subseteq K. \tag{9}$$

In fact, we have the following extension of Birkhoff's Theorem 10 (it is an extension in the sense that the one-to-one correspondence we obtain extends the correspondence in Birkhoff's Theorem). The result is due to Doignon and Falmagne (1985), who derive it from an appropriate 'Galois connection'.

Theorem 18. *There exists a one-to-one correspondence between the collection of all granular knowledge spaces $\mathcal{K}$ on a set Q and the collection of all the surmise functions σ on Q. One such correspondence is specified by the equivalence, for all q in Q and A in 2^Q,*

$$A \text{ is an atom at } q \text{ in } \mathcal{K} \quad \Longleftrightarrow \quad A \in \sigma(q). \tag{10}$$

This one-to-one correspondence links discriminative knowledge spaces to discriminative surmise functions.

The correspondence between knowledge spaces and surmise functions is suggestive of a practical method for building a knowledge space or even a learning space, based on analyzing large sets of learning data. We describe such a method in Section 11.

A characterization of learning spaces through their surmise functions derives directly from Theorem 15.

Theorem 19. *A finite knowledge space $(Q, \mathcal{K})$ is a learning space if and only if the corresponding surmise system (as in Theorem 18) has the property that any clause is a clause for only one item.*

In the case of finite, partially ordinal spaces, a highly efficient summary of the space takes the form of the 'Hasse diagram' of the partial order. Attempts to extend the notion of a Hasse diagram from partially ordered sets to surmise systems are reported in Doignon and Falmagne (1999) and Falmagne and Doignon (2011).

6 The Fringe Theorem

The final result of a standardized test is a numerical score[13]. In the case of an assessment in the framework of a learning space, the result is a knowledge state which may contain hundreds of items. Fortunately, a meaningful summary of that state can be given in the form of its 'inner fringe' and 'outer fringe'.

In the ten-item Example 4, consider the state $\{c, g, h, i, j\}$, which is printed in bold in the covering graph of Figure 2 (page 7). Figure 4 reproduces the relevant part of the graph, in particular all the adjacent states. From the state $\{c, g, h, i, j\}$, only three items are learnable[14], which are a, b and f (we mark them on their respective lines). On the other hand, the only way to reach state $\{c, g, h, i, j\}$ is to learn item j from the state $\{c, g, h, i\}$ (which is the unique state giving access to $\{c, g, h, i, j\}$). The two sets of items $\{j\}$ and $\{a, b, f\}$ completely specify the state $\{c, g, h, i, j\}$ among all the states in the structure[15]; this is a remarkable property of learning spaces which we now formalize.

Definition 20. Let $(Q, \mathcal{K})$ be a knowledge structure. The *inner fringe* of a state K in $\mathcal{K}$ is the set of items

$$K^{\mathcal{I}} = \{q \in K \mid K \setminus \{q\} \in \mathcal{K}\}. \tag{11}$$

The *outer fringe* of a state K is the set of items

$$K^{\mathcal{O}} = \{q \in Q \setminus K \mid K \cup \{q\} \in \mathcal{K}\}. \tag{12}$$

[13] Or a couple of such scores, in the case of a multidimensional model.

[14] We mean directly learnable without requiring the mastery of any other item outside $\{c, g, h, i, j\}$.

[15] In this particular case, the two-set summary is barely more concise than the original state. However, in realistic learning spaces containing millions of states, the summary may be considerably smaller than the state.

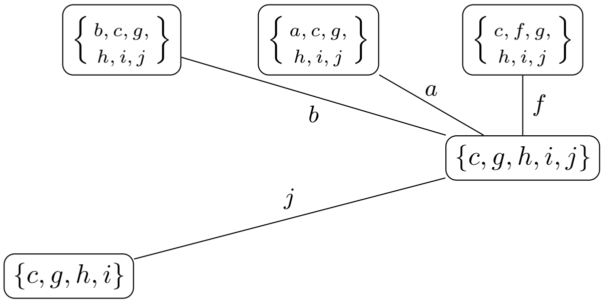
$\left\{ \begin{matrix} b, c, g, \\ h, i, j \end{matrix} \right\}$
$\left\{ \begin{matrix} a, c, g, \\ h, i, j \end{matrix} \right\}$
$\left\{ \begin{matrix} c, f, g, \\ h, i, j \end{matrix} \right\}$
a
f
b
$\{c, g, h, i, j\}$
j
$\{c, g, h, i\}$

Figure 4: Part of Figure 2 showing the items in the inner fringe and outer fringe of the state $\{c,\ g,\ h,\ i,\ j\}$ (see Definition 20).

Note that the empty state $\varnothing$ always has an empty inner fringe, and that the whole domain Q has an empty outer fringe.

Theorem 21. *In a learning space, any state is specified by the pair formed of its inner fringe and its outer fringe.*

Theorem 21 has the following important consequence. In any learning space, the knowledge state uncovered by an assessment can be reported as two sets of items: those in its inner fringe and those in its outer fringe. The outer fringe is especially important because, assuming that the learning space is a faithful representation of the cognitive organization of the material, it tells us exactly what the student is ready to learn. We will see in Section 12 that this information is accurate in real-life: the probability that a student actually succeeds in learning an item picked in the outer fringe of his or her state, estimated on the basis of hundreds of thousand ALEKS assessments, is about .93 (see page 45).

7 Learning Words and Learning strings

In a learning space, a learner can reach any state by learning its items one at a time—but not in any order. Let us look at an example.

Example 22. A learning space on the domain $Q = \{a, b, c, d\}$ is described by its covering diagram in Figure 5.

The state $\{a,\ b,\ d\}$ can be reached by mastering the items in three possible successions, which we also call ‘words’ (see Definition 23):

$$a\,b\,d,$$
$$b\,a\,d,$$
$$b\,d\,a.$$

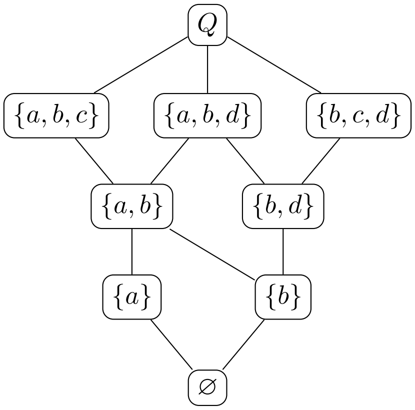

Figure 5: The covering diagram of the learning space in Example 22.

For the mastery of the whole domain Q, there are 6 'strings' in all:

$$\begin{gathered} a\,b\,c\,d, \\ a\,b\,d\,c, \\ b\,a\,c\,d, \\ b\,a\,d\,c, \\ b\,d\,a\,c, \\ b\,d\,c\,a. \end{gathered}$$

All of the words and strings in Example 22 share a self-explanatory property: for any of their 'prefixes', the items appearing in the prefix form a knowledge state. Let us define the new terminology.

Definition 23. Given some finite set Q, a *word* on Q is any injective mapping f from $\{1,\ 2,\ \ldots,\ k\}$ to Q, for some k with $0 \leq k \leq |Q|$; the case $k = 0$ produces the *empty word*. With $f(i) = w_i$ for $1 \leq i \leq k$, we write the word f as $w = w_1\, w_2 \cdots w_k$, and we call k the *length* of the word w. A *prefix* of w is a word $w_1\, w_2 \cdots w_i$, where $0 \leq i \leq k$ (thus any word is a prefix of itself). If an item q does not appear in w, the *concatenation* of w with q is the word $w\, q = w_1\, w_2 \cdots w_k\, q$. A *string* on Q is a word of length $|Q|$. Notice that our words (and strings) do not involve repetitions (because of the required injectivity of f), a reason for which some authors rather speak of "simple words" (as for example Boyd and Faigle, 1990). Any word $w = w_1\, w_2 \cdots w_k$ *determines* the set $\widetilde{w} = \{w_1,\ w_2,\ \ldots,\ w_k\}$ (if all words are counted, $k!$ of them determine the same set of size k).

Let $(Q, \mathcal{K})$ be a finite knowledge structure with $|Q| = m$. A *learning word* (in $(Q, \mathcal{K})$) is a word w on Q such that for each of its prefixes $v = w_1\, w_2 \cdots w_i$

the subset $\widetilde{v} = \{w_1, w_2, \ldots, w_i\}$ is a state in $\mathcal{K}$; here $0 \leq i \leq k$ if k is the length of w. A *learning string* is such a learning word with $k = m$.

Korte et al. (1991) use the expression "shelling sequence" for our "learning word", and the expression "basic word" for our "learning string", while Eppstein (2013a) uses "learning sequence" for "learning string".

General knowledge spaces can be without any learning string (for instance, it is the case when $\mathcal{K} = \{\varnothing, Q\}$ as soon as $|Q| \geq 2$). The axioms of learning spaces are consistent with the existence of (many) learning strings and words. In fact, learning spaces can be recognized from properties of the collection of their learning strings (see next theorem) or the collection of their learning words (see Theorem 26).

Theorem 24. *Let Q be a finite set, with $|Q| = m$. A nonempty collection $\mathcal{S}$ of strings on Q is the collection of all learning strings of some learning space on Q if and only if $\mathcal{S}$ satisfies the three conditions below:*

(i) *any item appears in some string of $\mathcal{S}$;*

(ii) *if u and v are two strings in $\mathcal{S}$ such that for some k in $\{1, 2, \ldots, m-1\}$ we have*

$$\{\, u_1, u_2, \ldots, u_{k-1} \,\} = \{\, v_1, v_2, \ldots, v_{k-1} \,\} \quad \text{and} \quad u_k \neq v_k, \tag{13}$$

then

$$u_1\, u_2\, \cdots\, u_{k-1}\, u_k\, v_k \tag{14}$$

is a prefix of some string in $\mathcal{S}$;

(iii) *if u and v are two strings in $\mathcal{S}$ such that for some k in $\{0, 1, \ldots, m-1\}$ and some item q we have*

$$\{v_1, v_2, \ldots, v_{k-1}, v_k, v_{k+1}\} \setminus \{u_1, u_2, \ldots, u_k\} = \{q\}, \tag{15}$$

then $u_1\, u_2\, \cdots\, u_k\, q$ is a prefix of some string in $\mathcal{S}$.

Proof. (Necessity.) Assume $(Q, \mathcal{L})$ is a learning space with $|Q| = m$, and denote by $\mathcal{S}$ the collection of its learning strings. By Learning Smoothness $\mathcal{S}$ is nonempty and each item of Q appears in some string in $\mathcal{S}$, so Condition (i) is true. If the hypothesis of Condition (ii) holds, then $\{u_1, u_2, \ldots, u_{k-1}\}$, $\{u_1, u_2, \ldots, u_{k-1}, u_k\}$ and $\{u_1, u_2, \ldots, u_{k-1}, v_k\}$ are all states of $\mathcal{L}$. Hence by Learning Consistency $\{u_1, u_2, \ldots, u_{k-1}, u_k, v_k\}$ is also a state, which we denote by L. On the other hand, by Learning Smoothness, there is a sequence L_{k+1}, L_{k+2}, $\ldots$, L_m of states with $L_{k+1} = L$, $L_m = Q$, and $|L_i \setminus L_{i-1}| = 1$ for $i = k+2$, $k+3, \ldots, m$. Taking w_i as the item in $L_i \setminus L_{i-1}$, we obtain the learning string $u_1\, u_2\, \cdots\, u_{k-1}\, u_k\, v_k\, w_{k+2}\, w_{k+3}\, \cdots\, w_m$. Thus Condition (ii) holds.

Now suppose the strings u and v fulfil the assumption in Condition (iii). Thus $\{v_1, v_2, \ldots, v_{k+1}\}$ is a state, that we call L. By Learning Smoothness, there is a sequence L_{k+1}, L_{k+2}, $\ldots$, L_m of states in $\mathcal{L}$ with $L_{k+1} = L$, $L_m = Q$

and $|L_i \setminus L_{i-1}| = 1$ for $i = k+2,\ k+3,\ \ldots,\ m$. With $\{w_i\} = L_i \setminus L_{i-1}$, we obtain the learning string

$$u_1\, u_2\, \cdots\, u_{k-1}\, u_k\, q\, w_{k+2}\, w_{k+3}\, \cdots\, w_m.$$

Hence Condition (iii) holds.

(Sufficency.) Given a collection $\mathcal{S}$ of strings on Q satisfying (i)–(iii), we call $\mathcal{L}$ the collection of all prefixes of strings in $\mathcal{S}$. Then $\varnothing$ and Q are in $\mathcal{L}$. Moreover, $\mathcal{L}$ clearly satisfies downgradability, that is Condition (b) in Theorem 8. To establish that $\mathcal{L}$ satisfies Condition (c), let K, $K \cup \{q\}$ and $K \cup \{r\}$ be in $\mathcal{L}$. There exist strings u and v such that $\{u_1,\ u_2,\ \ldots,\ u_{|K|}\} = K$ and $\{v_1,\ v_2,\ \ldots,\ v_{|K|+1}\} = K \cup \{q\}$. Then by Condition (iii) $u_1\, u_2\, \cdots\, u_{|K|}\, q$ is a prefix of some string in $\mathcal{S}$. A similar argument shows that $u_1\, u_2\, \cdots\, u_{|K|}\, r$ is a string prefix. Then by Condition (ii), $u_1\, u_2\, \cdots\, u_{|K|}\, q\, r$ is also a prefix of some string. So $K \cup \{q, r\} \in \mathcal{L}$. Hence, by Theorem 8, $(Q, \mathcal{L})$ is a learning space. Moreover, the learning strings of $(Q, \mathcal{L})$ constitute exactly $\mathcal{S}$ (this derives again from the definition of $\mathcal{S}$ together with Condition (iii)). □

Here is an example showing that Conditions (ii) and (iii) in Theorem 24 are independent.

Example 25. On the domain $Q = \{a, b, c, d\}$, the two strings

$$a\, b\, d\, c,$$
$$a\, c\, d\, b$$

form a collection which satisfies Condition (iii) in Theorem 24 but not Condition (ii) (take $k = 2$, $u_1\, u_2 = a\, b$ and $v_1\, v_2 = a\, c$). Conversely, the two strings on the same domain $Q = \{a, b, c, d\}$

$$a\, b\, c\, d,$$
$$b\, a\, d\, c$$

form a collection which satisfies Condition (ii) in Theorem 24 but not Condition (iii) (take $k = 2$, $u_1\, u_2 = b\, a$ and $v_1\, v_2\, v_3 = a\, b\, c$).

The next result is Theorem 2.1 in Boyd and Faigle (1990) (compare with Theorem 1.4 in Korte et al., 1991).

Theorem 26. *Let Q be a finite domain. A collection $\mathcal{W}$ of words on Q is the collection of all learning words of some learning space on Q if and only if $\mathcal{W}$ satisfies the following three conditions:*

(i) *any item from Q appears in at least one word of $\mathcal{W}$;*

(ii) *any prefix of a word in $\mathcal{W}$ also belongs to $\mathcal{W}$;*

(iii) *if v and w are two words of $\mathcal{W}$ with $\widetilde{v} \not\subseteq \widetilde{w}$, then for some item q in $\widetilde{v} \setminus \widetilde{w}$ the concatenation $w\, q$ is a word again in $\mathcal{W}$.*

Proof. (Necessity) Assume $(Q, \mathcal{L})$ is a learning space, and denote by $\mathcal{W}$ the collection of all its learning words. Then by downgradability of $\mathcal{L}$ and $Q \in \mathcal{L}$, the collection $\mathcal{W}$ contains some string, so Condition (i) is true. By the definition of a learning word w, any prefix of w is also a learning word, so Condition (ii) holds. Now take two words v and w as in Condition (iii). Then $\widetilde{w} \cup \widetilde{v} \in \mathcal{L}$ (by Theorem 7, $\mathcal{L}$ is $\cup$-closed). Because of Learning Smoothness, there is a sequence $L_0 = \widetilde{w}$, L_1, …, $L_\ell = \widetilde{w} \cup \widetilde{v}$ with $|L_i \setminus L_{i-1}| = 1$ for $i = 1, 2, \ldots, \ell$. Let $\{q\} = L_1 \setminus L_0$. Then $q \in \widetilde{v} \setminus \widetilde{w}$ and $w\,q \in \mathcal{W}$.

(Sufficiency) Given a collection $\mathcal{W}$ of words as in the statement, set $\mathcal{L} = \{\widetilde{w} \mid w \in \mathcal{W}\}$. Then $\varnothing \in \mathcal{L}$. Repeatedly applying Conditions (i) and (iii), we infer that there is a string in $\mathcal{W}$, and so $Q \in \mathcal{L}$. By (ii), $\mathcal{L}$ is downgradable. To conclude that $(Q, \mathcal{L})$ is a learning space it now suffices to prove that $\mathcal{L}$ satisfies Condition (iii) in Theorem 8. Let K, $K \cup \{q\}$ and $K \cup \{r\}$ be states in $\mathcal{L}$ with $q \neq r$. There are then some words v and w in $\mathcal{W}$ such that $\widetilde{v} = K \cup \{q\}$ and $\widetilde{w} = K \cup \{r\}$. Because $\widetilde{v} \setminus \widetilde{w} = \{q\}$, Condition (iii) implies $w\,q \in \mathcal{W}$. Now $\widetilde{w\,q} = K \cup \{q, r\}$, and so $K \cup \{q, r\} \in \mathcal{L}$. Finally, it is easily checked that $\mathcal{W}$ consists of all learning words of $\mathcal{L}$. □

Learning words and strings form a useful tool for the handling of large learning spaces. For instance, they are implicit in the new representation in Figure 6 of the learning space $\mathcal{L}$ from Example 22: a learning string consists of the letters (representing items) on a path from the vertex representing $\varnothing$ to the vertex representing Q. We call such a representation (with letters displayed only to show addition of a single item) a *learning diagram*.

Figure 7 on page 22 shows a similar learning diagram for our ten-item example from Example 4.

Theorem 24 characterizes learning spaces through their complete collections of learning strings. In the same vein as the base which, containing only a relatively small number of states of the knowledge structure, gives us access to the whole collection, we might want to summarize in a similar way the collection of learning strings in a subcollection. The following definition comes from Eppstein (2013a).

Definition 27. Let $\mathcal{S}$ be a collection of strings on a finite domain Q. Form the collection $\mathcal{L}_\mathcal{S}$ containing all possible unions of the sets determined by prefixes of strings in $\mathcal{S}$. Then $(Q, \mathcal{L}_\mathcal{S})$ is the learning space *encoded* by $\mathcal{S}$.

That $(Q, \mathcal{L}_\mathcal{S})$ indeed forms a learning space is easy to verify (for instance, apply Theorem 7(ii)). Now, conversely, given a learning space $(Q, \mathcal{L})$, we may select any nonempty subset $\mathcal{S}$ of its collection of learning strings; then, in general, $\mathcal{L}_\mathcal{S} \subseteq \mathcal{L}$ holds, but there is no reason to have equality here (see Theorem 13.5.7 in Eppstein, 2013a, for a criterion).

The case in which we have the equality $\mathcal{L}_\mathcal{S} = \mathcal{L}$ is interesting for algorithmic work on the learning space $\mathcal{L}$. Let us denote as S_1, S_2, …, S_k the strings forming $\mathcal{S}$. Then any state in $\mathcal{L}$ is univocally encoded by a list of natural numbers n_1, n_2, …, n_k: each of the numbers specifies the length of the prefix we need to

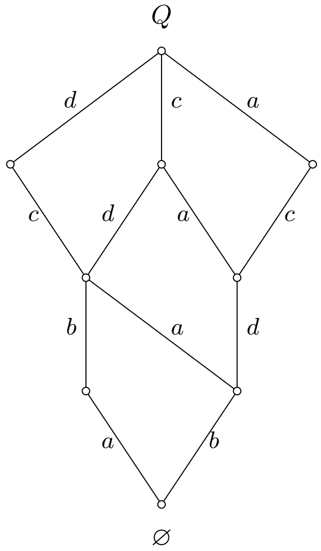

Figure 6: The learning diagram representing the learning space in Example 22 (see also Figure 5).

extract from the corresponding string in $\mathcal{S}$ to get the state at hand as the union of the prefixes. Eppstein (2013a) shows how to exploit the new state encoding for various tasks. The present context generates the following problem: given a learning space, how do we compute the smallest number of learning strings needed to encode it? The ensuing invariant was dubbed *convex dimension* by Edelman and Jamison (1985) (see also Korte et al., 1991). Eppstein (2013b) gives an algorithm to compute it.

Example 28. The learning space of Example 22 (see also Figure 5) is encoded by the following three of its learning strings:

$$a\,b\,d\,c,$$
$$b\,a\,c\,d,$$
$$b\,d\,c\,a.$$

It is as well encoded by the two strings

$$a\,b\,c\,d,$$
$$b\,d\,c\,a,$$

but never by just one string.

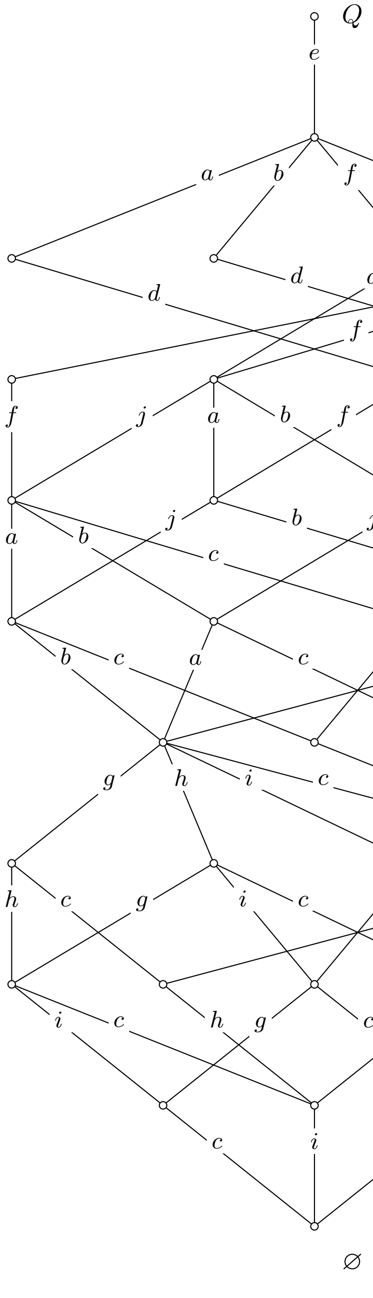

Figure 7: The learning diagram of the ten-item learning space $\mathcal{L}$ of Example 4 (see also Figure 6).

8 The Projection Theorem

How large is the structure of a real-life learning space? For instance, what is the ratio of the number of knowledge states to the number of possible subsets of the domain? In the ten-item example of Table 1 and Figure 2 we have 34 knowledge states, which gives the ratio $\frac{34}{2^{10}} \approx .03$. However, this example may be misleading. In real-life learning spaces, the ratio may become considerably smaller as soon as there are a couple of dozen items. As another illustration, we again take the 37 items example in Beginning Algebra. There are 4 615 knowledge states in the corresponding (induced) learning space. With just 37 items, the ratio of the number of states to the number of subsets is $\frac{4\,615}{2^{37}} \approx .03 \times 10^{-6}$. We mention in passing that there are 217 different knowledge states containing exactly 25 items. Presumably all these knowledge states would be assigned the same psychometric score in classical test theory, while KST treats them as different.

As mentioned in Section 1, the full domain of Beginning Algebra in the ALEKS system contains about 650 items. The complexity of the resulting learning space is daunting. It calls for ways of parsing such huge learning spaces into meaningful components. One of the goals could be a placement test for which only part of the full collection of items is needed. A more important reason arises when we need an assessment on the full structure. In such a case, the number of knowledge states is so large that a straightforward approach becomes infeasible.

A practical solution has been worked out which consists in suitably partitioning the domain and then carrying on simultaneous, parallel, mutually informative assessments of the resulting substructures, ultimately followed by the computation of the final state. We outline this technique in Section 10. Here, we define two useful concepts, that of a 'projection' and that of 'children' of a learning space, given a subset of the domain. We show—without proof—that such a projection remains a learning space, while in general children satisfy only some of the requirements in Definition 3.

We begin with an illustration based on a small learning space.

Example 29. Let $(Q, \mathcal{K})$ be the learning space on the domain $\{a, b, c, d\}$ whose covering diagram is provided in Figure 8. Consider the subset $Q' = \{c, d\}$ of the domain Q and form all the 'traces' $K \cap Q'$, for K any state in $\mathcal{K}$. The resulting collection, the 'projection' of $\mathcal{K}$ on Q',

$$\{\, \varnothing,\ \{d\},\ \{c, d\}\,\}, \tag{16}$$

forms again a learning space. The general result appears in Theorem 33(i) below. We summarize the construction in Figure 9.

With the same space we now illustrate another construction. This time, we sort out the states in $\mathcal{K}$ according to their intersections with $Q' = \{c, d\}$. The resulting equivalence classes are displayed row by row in Table 4. Thus the states of $\mathcal{K}$ in a same row all have the same trace on $Q' = \{c, d\}$ (shown in the second column). It happens that they always form a 'union-stable', well-graded knowledge structure (see Theorem 33(ii) below). However, in view of

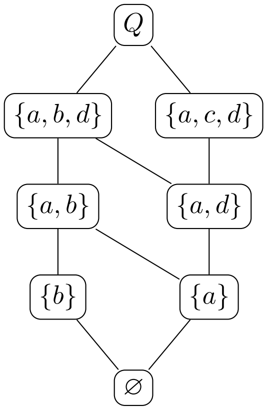

Figure 8: The covering diagram of the learning space in Example 29.

the absence of $\varnothing$, the two last structures do not constitute a learning space. In the third column, we show the 'children'; they are obtained by subtracting from the states (of $\mathcal{K}$ shown in that row) their common intersection.

Table 4: The states of the learning space $(Q, \mathcal{K})$ in Example 29 are sorted according to their intersection with $\{c, d\}$. The third column provides the corresponding children.

Classes of states	Intersections with $\{c, d\}$	Children
$\{\{a, c, d\},\ Q\}$	$\{c, d\}$	$\{\varnothing,\ \{b\}\}$
$\{\{a, d\},\ \{a, b, d\}\}$	$\{d\}$	$\{\varnothing,\ \{b\}\}$
$\{\varnothing,\ \{a\},\ \{b\},\ \{a, b\}\}$	$\varnothing$	$\{\varnothing,\ \{a\},\ \{b\},\ \{a, b\}\}$

In the second row of Table 4, all the states (of the original learning space) have in common the items d (by construction) and moreover a. Removing the two common items gives the two sets $\varnothing$ and $\{b\}$ which altogether form a learning space on the new domain $\{b\}$. The last assertion is not true in general.

Example 30. Let $(Q, \mathcal{K})$ be the learning space on the domain $\{a, b, c, d\}$ whose covering diagram is provided in Figure 10. For $Q' = \{a\}$, one of the two children equals $\{\{c\},\ \{d\},\ \{c, d\}\}$; it does not contain the empty set.

We now define the concept of a projection, and then that of children.

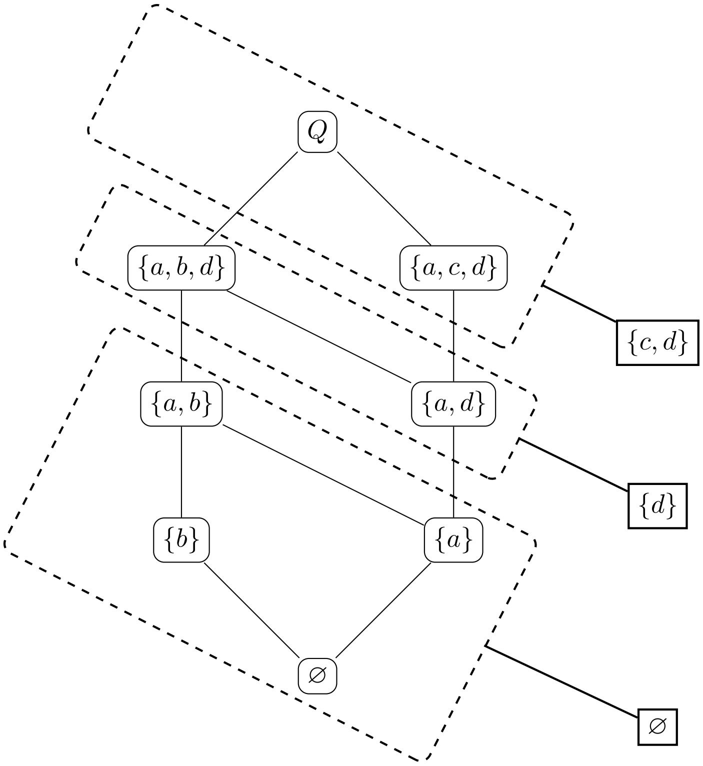

Figure 9: An illustration of the two constructions in Example 29 (with $Q' = \{c, d\}$): the three equivalence classes are in the rounded, dashed rectangles; the traces are displayed on the right.

Definition 31. Suppose that $(Q, \mathcal{K})$ is a knowledge structure with $|Q| \geq 2$, and let Q' be any proper nonempty subset of Q. For K in $\mathcal{K}$, the subset $K \cap Q'$ of Q' is the *trace* of K on Q'. The collection of all traces

$$\mathcal{K}_{|Q'} = \{K \cap Q' \mid K \in \mathcal{K}\} \tag{17}$$

is the *projection* of $\mathcal{K}$ on Q'.

Figure 11 shows the trace of the ten-item learning space on $\{c, d, g, j\}$. Note that the sets in $\mathcal{K}_{|Q'}$ may be or not be states of $\mathcal{K}$.

Definition 32. Suppose again that $(Q, \mathcal{K})$ is a knowledge structure with $|Q| \geq 2$, and let Q' be any proper nonempty subset of Q. Define the relation $\sim_{Q'}$ on $\mathcal{K}$ by

$$K \sim_{Q'} L \iff K \cap Q' = L \cap Q' \tag{18}$$

$$\iff K \triangle L \subseteq Q \setminus Q'. \tag{19}$$

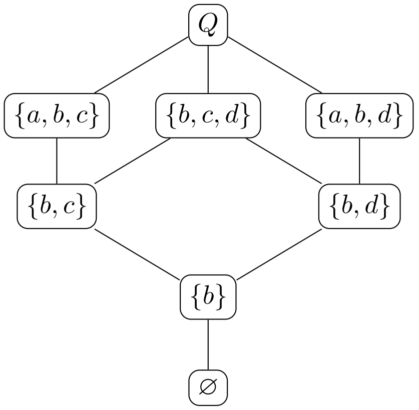

Figure 10: The covering diagram of the learning space in Example 30.

(The equivalence between the right hand sides of (18) and (19) is easily verified.) Then $\sim_{Q'}$ is an equivalence relation on $\mathcal{K}$. When the context specifies the subset Q', we may simply write $\sim$ for $\sim_{Q'}$. We denote by $[K]$ the equivalence class of $\sim$ containing K, and by $\mathcal{K}_\sim = \{\, [K] \;\big|\; K \in \mathcal{K}\}$ the partition of $\mathcal{K}$ induced by $\sim$ (or by Q'). For any state K in $\mathcal{K}$, we define the collection

$$\mathcal{K}_{[K]} = \{\, L \setminus \cap[K] \;\big|\; L \in [K]\}. \tag{20}$$

The collection $\mathcal{K}_{[K]}$ is a *Q'-child* of $\mathcal{K}$, or simply a *child* of $\mathcal{K}$ when the set Q' is made clear by the context. A child of $\mathcal{K}$ may take the form of the singleton $\{\varnothing\}$; it is then the *trivial* child. We refer to $\mathcal{K}$ as the *parent* structure.

Because $\varnothing \in \mathcal{K}$ we have $\mathcal{K}_{[\varnothing]} = [\varnothing]$. We may have $\mathcal{K}_{[K]} = \mathcal{K}_{[L]}$ even when $K \not\sim L$. (Examples are easily built: see Table 4.)

Theorem 33. *Let $(Q, \mathcal{K})$ be a learning space, with $|Q| \geq 2$. The following two properties hold for any proper nonempty subset Q' of Q.*

(i) *The projection $\mathcal{K}_{|Q'}$ of $\mathcal{K}$ on Q' is a learning space.*

(ii) *The children of $\mathcal{K}$ are well-graded and $\cup$-stable collections. The latter means that the union of any nonempty subcollection of the collection also belongs to the collection*[16]*. Example 29 shows that the children are not necessarily learning spaces.*

We can impose some restricting conditions on the set Q' that guarantee any child to be a learning space, provided that the empty state is added to the collection if necessary (see Falmagne and Doignon, 2011, Definition 2.4.11 and Theorem 2.4.12).

[16] Notice a slight difference between "union-stable" and "union-closed"; only the second condition entails that the empty set belongs to the collection.

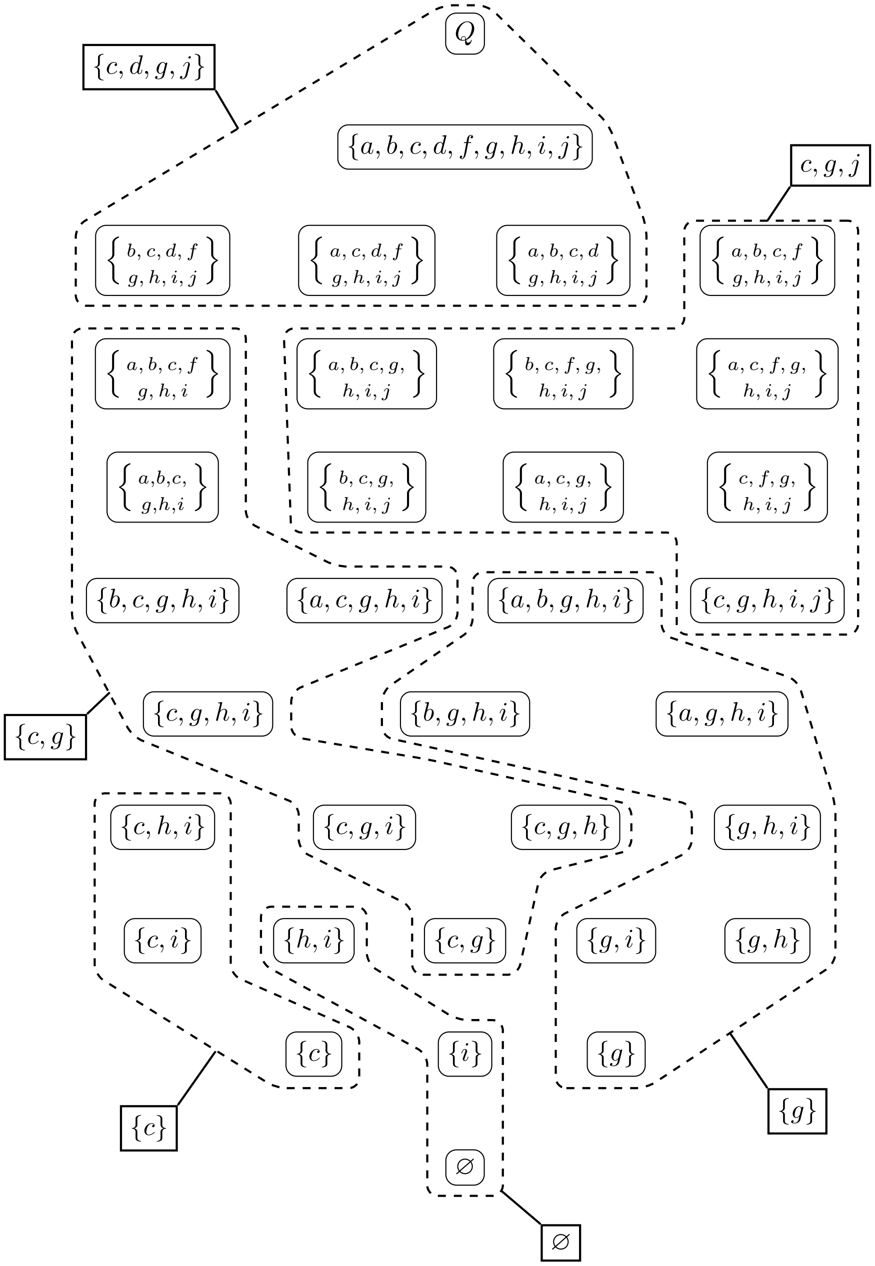

Figure 11: Projection of the ten-item example on $\{c, d, g, j\}$: the dashed lines delineate the equivalence classes, the little black rectangles show the traces.

The concept of a projection plays an essential role in designing assessment algorithms for realistic learning spaces. In applications, the size of the collection of states may be so prohibitively large that the obvious strategy of gradually narrowing down, by some method or other, the class of states consistent with the assessment results is not practical. The solution discussed in Subsection 10.1 is to first design a suitable partition of the domain into N manageable classes. Second, one builds the N projections on these classes. The assessment procedure then operates in parallel on the N projections. A combination of the results of the N simultaneous assessments delivers a final knowledge state.

9 Probabilistic Knowledge Structures

The concept of a learning space is deterministic. As such, it does not provide realistic predictions of subjects' responses to the problems of a test. Probabilities must enter in at least two ways in a realistic model. For one, the knowledge states will certainly occur with different frequencies in the population of reference. So, it makes sense to postulate the existence of a probability distribution on the collection of states. For another, a subject's knowledge state does not necessarily specify the observed responses. A subject having mastered an item may be careless in responding, and make an error. Also, in some situations, a subject may be able to guess the correct response to a question not yet mastered[17]. In general, it makes sense to introduce conditional probabilities of responses, given the states.

Definition 34. A *probabilistic (knowledge structure)* $(Q, \mathcal{K}, p)$ consists of a finite knowledge structure $(Q, \mathcal{K})$ with a probability distribution p on the collection $\mathcal{K}$ of states. Thus $p(K)$ is a real number with $0 \leq p(K) \leq 1$, for any state K, and moreover $\sum_{K \in \mathcal{K}} p(K) = 1$. A *parametrized (probabilistic knowledge structure)* $(Q, \mathcal{K}, p, \beta, \eta)$ is a probabilistic knowledge structure $(Q, \mathcal{K}, p)$ equipped with two functions $\beta : Q \to [0, 1] : q \mapsto \beta_q$ and $\eta : Q \to [0, 1] : q \mapsto \eta_q$. The number β_q represents the *careless error probability* to item q, and the number η_q is the *lucky guess probability* for item q.

Definition 35. A parametrized probabilistic knowledge structure $(Q, \mathcal{K}, p, \beta, \eta)$ is *straight* if $\beta_q = \eta_q = 0$ for all items q.

We now describe two types of construction of probability distributions on a set of states. In the first type, the set of states results from a projection.

Definition 36. Let $(Q, \mathcal{K}, p)$ be a probabilistic knowledge structure, and let $\varnothing \neq Q' \subset Q$. On the projection $\mathcal{K}_{|Q'}$, we define the *projected distribution* p' by setting, for K' a state in $\mathcal{K}_{|Q'}$,

$$p'(K') \;=\; \sum \{p(K) \,\big|\, K \in \mathcal{K} \text{ and } K \cap Q' = K'\}\,.$$

[17] Such lucky guesses have probability zero or are very unlikely in assessment systems requiring open responses to the items, instead of multiple-choice. ALEKS is one of those systems.

Then $(Q', \mathcal{K}_{Q'}, p')$ is the *probabilistic projection* on Q' of the probabilistic knowledge structure $(Q, \mathcal{K}, p)$.

In the second case, we start with a probabilistic knowledge structure and extend it on a larger set of states.

Definition 37. Let $(Q, \mathcal{K})$ be a knowledge structure, and let $\varnothing \neq Q' \subset Q$. Assume p' is a probability distribution on the projection $\mathcal{K}_{|Q'}$. We define the *extended distribution* p^+ to $\mathcal{K}$ of p by setting, for K a state in $\mathcal{K}$,

$$p^+(K) \;=\; \frac{p'(K \cap Q')}{\sum |\{L \in \mathcal{K} \,\big|\, L \cap Q' = K \cap Q'\}|}.$$

Then $(Q, \mathcal{K}, p^+)$ is the *uniform extension* to $(Q, \mathcal{K})$ of the probabilistic knowledge structure $(Q', \mathcal{K}', p')$.

10 The Stochastic Assessment Algorithm

The general idea of the assessment algorithm is to gradually update, after each response of the subject, the distribution of probabilities on the collection of states. On each step of the assessment, the system selects an item, and presents a randomly chosen instance of that item to the student. The student's response is evaluated and classified as "correct" or "false". The result serves to update the probability distribution on the set of states. The new distribution is the starting point of the next step. Ultimately, only one or a few states will remain with a high probability. The system then chooses the final state.

The assessment algorithm we just sketched is applicable in the 'straightforward situation', that is, when the learning space (or the knowledge structure for that matter) is moderately large, with a domain not exceeding 50 items. Such learning spaces can serve in the design of some placement tests, for example. In Subsection 10.2, we deal with the more usual case of domains having hundreds of items. On such large domains, the application of the algorithm requires considerably more sophistication because the number of states becomes so large that operating on the probability distribution on the set of states is unmanageable. The solution outlined in Subsection 10.2 is to build a suitable partition of the domain and to perform parallel (simultaneous, mutually informative) assessments on the projections on all the subdomains. The final state is constructed by combining the outcomes obtained in each of the parallel assessments.

10.1 Sketch of the algorithm in the straightforward situation

We suppose that, at the outset of the algorithm execution, there exists some probability distribution on the collection of states[18]. Such a probability distribution may be inferred from some information on the population that the

[18] This is one of the reasons why the algorithm cannot be readily applied in the case of large domains: we cannot manage a probability distribution on a set containing billions of states.

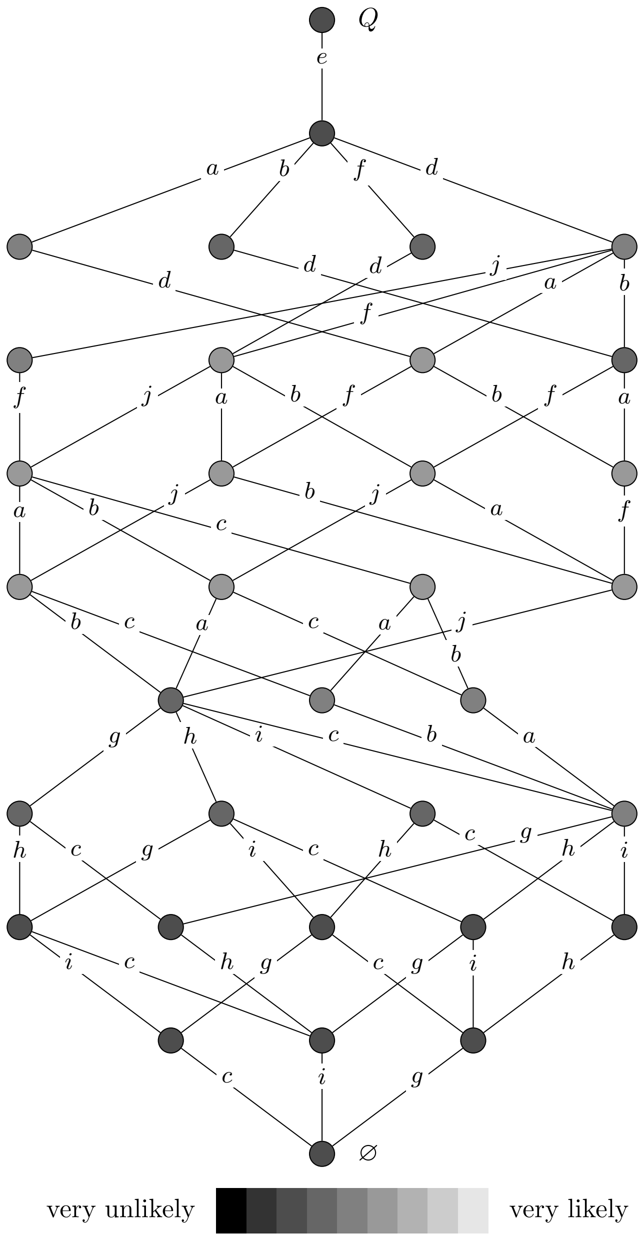

Figure 12: Initial probabilities of all the states at the beginning of the assessment. The probabilities are marked by the shading of the circles representing the states. Dark grey means very unlikely, and bright grey very likely.

testee belongs to. Failing such information, we may simply assume that the distribution is uniform. The assessment is adaptive. On each of its main steps, the algorithm applies a Bayesian type updating operator of the current probability distribution, producing thus for each knowledge state an estimate of the probability that the student be in this state.

We keep illustrating our discussion by our example of the 10 items in Beginning Algebra and give in Figure 12 a picture of the learning space at the beginning of the assessment. Notice that the probability values for the states are suggested by the strength of gray (the lighter the disk is, the higher the probability value is).

Figure 13 summarizes the sequence of events on each step. We call such steps *trials*. At the beginning of each trial, the algorithm picks the 'most informative item'[19] to give the student; for doing so, it relies on the current, estimated probabilities of the states. Next, it randomly chooses an instance of the item. For example, if the item selected is

a : Quotients of expressions involving exponents

the instance could be

$$\text{Simplify the expression} \quad \frac{a^4b^5}{5a^6b} \quad \text{as much as possible.}$$

The algorithm then proposes the instance to the student. It records the response and checks the correctness. To complete the trial, the algorithm uses the information just collected in order to update the probability distribution on the collection of states. Of course, we still need to specify how the item selection and the probability updating are performed (see the Questioning Rule and the Updating Rule below, page 34).

The basic idea of the assessment algorithm is to ensure that, in the course of the assessment, the probability distribution becomes gradually concentrated on a knowledge state, or on a few knowledge states which are close together[20]. If several states end up with the same high probability, the system chooses randomly between them. In practical situations, the assessment terminates in about 25–35 trials with the algorithm we will describe. We first provide an illustration and then introduce the required notation.

Figure 12 pictured a (possible) probability distribution on the set of states at the beginning of the assessment. The brightest shades indicate the high probability states, the darkest ones represent the lowest probability states. We infer from the picture that the student may have master about 4–7 items. On page 33, Figure 14 shows the graphs of three later situations. Suppose the student gives during the first trial a correct response to item a. The graph on the left shows the updated probability values: note the increase of the probability values for states containing a, and the decrease of the values for the states not containing a. The middle graph sketches the estimated distribution at the end

[19] For the technical meaning of this term, see the Questioning Rule on page 34.

[20] From the standpoint of their symmetric difference.

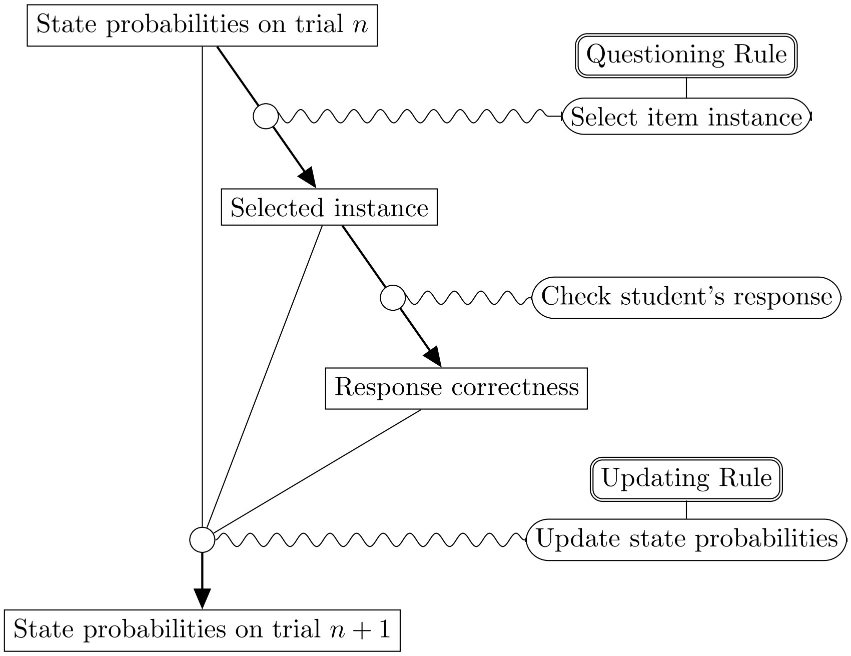

Figure 13: Top to bottom, a schematic representation of the assessment from trial n to trial $n+1$: on the left the data acted upon, on the right the general instructions executed according to specific rules.

of trial 2, after the student gave next a false response to item f. The last graph represents the typical situation at the end of the assessment: only one state remains with a high probability, which is here $\{a, c, g, h, i, j\}$. The algorithm would choose that state as representing the competence of the student in that part of Beginning Algebra.

We now give a precise mathematical meaning to the concepts outlined above, starting with a list of notations based on a knowledge structure $(Q, \mathcal{K})$.

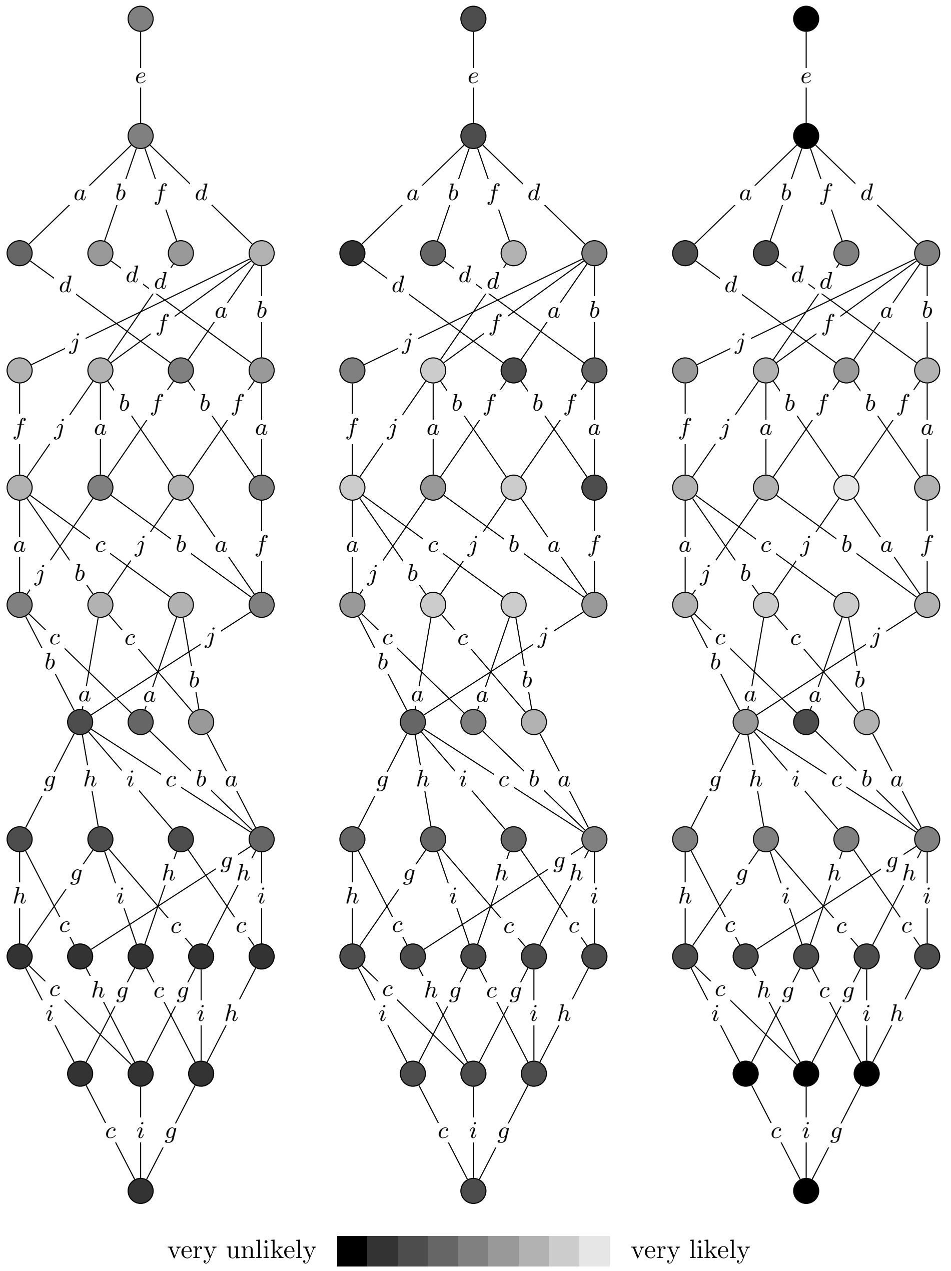

Figure 14: Left to right, three successive situations along the assessment: (i) after a correct response to a; (ii) next, after a false answer to f; (iii) later, at the end of the assessment.

Mathematical notation.

n the step number, or *trial number*, with $n = 1, 2, \ldots;$

$\mathcal{K}_q$ the subcollection of $\mathcal{K}$ formed by the states containing q;

Λ_+ the set of all positive probability distributions on $\mathcal{K}$;

$\mathbf{L}_n$ a positive random probability distribution on $\mathcal{K}$: we have $\mathbf{L}_n = L_n \in \Lambda_+$ if L_n is the probability distribution on $\mathcal{K}$ at the beginning of trial n (so $L_n > 0$);

$\mathbf{L}_n(K)$ a random variable (r.v.) evaluating the probability of state K on trial n;

$\mathbf{Q}_n$ a r.v. representing the question asked on trial n: we have $\mathbf{Q}_n = q$ if q in Q is the question asked on trial n;

$\mathbf{R}_n$ a r.v. coding the response on trial n:
$$\mathbf{R}_n = \begin{cases} 1 & \text{if the response is correct,} \\ 0 & \text{otherwise;} \end{cases}$$

$\mathbf{W}_n$ the random history of the process from trial 1 to trial n;

ι_A the indicator function of a set A: $\iota_A(q) = \begin{cases} 1 & \text{if } q \in A, \\ 0 & \text{if } q \notin A; \end{cases}$

$\zeta_{q,r}$ a collection of parameters defined for $q \in Q$, $r \in \{0, 1\}$ and satisfying $1 < \zeta_{q,r}$ (see Updating Rule [U] below).

The formal rules governing the assessment algorithm are as follows (where n is the trial number).

Using the 'Questioning Rule', the algorithm picks a most informative item, that is, an item which, on the basis of the current probability distribution on the set of states, has a probability of being responded to correctly as close to .5 as possible. There may be more than one such item, in which case a uniform, random selection is made. Formally, in terms of the actual probability distribution L_n:

[Q] **Questioning Rule.** For all $q \in Q$ and all positive integers,

$$\mathbb{P}(\mathbf{Q}_n = q \mid \mathbf{L}_n, \mathbf{W}_{n-1}) = \frac{\iota_{S(L_n)}(q)}{|S(L_n)|} \tag{21}$$

where $S(L_n)$ is the subset of Q containing all those items q minimizing $|2L_n(\mathcal{K}_q) - 1|$. Under this questioning rule, which is called *half-split*, we must have $\mathbf{Q}_n \in S(L_n)$ with a probability equal to one. The questions in the set $S(L_n)$ are then chosen with equal probability.

The 'Response Rule' formalized below states that the student's response to an instance of an item is correct with probability 1 if the item belongs to the student's knowledge state K_0, and false with probability 1 otherwise. While this rule plays no role in the assessment algorithm per se, it is essential in some simulation, and so is its elaborated version in the form of the 'Modified Response

Rule' (see page 36 below). These rules are used, in particular, in the 'Extra problem method' (see Subsection 0.18.1, page 43).

[R] **Response Rule.** There is some state K_0 such that, for all positive integers n,

$$\mathbb{P}(\mathbf{R}_n = \iota_{K_0}(q) \bigm| \mathbf{Q}_n = q, \mathbf{L}_n, \mathbf{W}_{n-1}) = 1\,. \tag{22}$$

The state K_0 is the *latent state* representing the set of all the items currently mastered by the student. It is the state that the assessment algorithm aims to uncover.

[U] **Updating Rule.** We have $\mathbb{P}(\mathbf{L}_1 = L) = 1$ (with L the initial probability distribution). Moreover there are real parameters $\zeta_{q,r}$ (where $q \in Q$, $r \in \{0, 1\}$ and $1 < \zeta_{q,r}$) such that for any positive integer n, if $\mathbf{L}_n = L_n$, $\mathbf{Q}_n = q$, $\mathbf{R}_n = r$ and

$$\zeta^K_{q,r} = \begin{cases} 1 & \text{if } \iota_K(q) \neq r, \\ \zeta_{q,r} & \text{if } \iota_K(q) = r, \end{cases} \tag{23}$$

then

$$L_{n+1}(K) \;=\; \frac{\zeta^K_{q,r} L_n(K)}{\sum_{K' \in \mathcal{K}} \zeta^{K'}_{q,r} L_n(K')}. \tag{24}$$

This updating rule is called *multiplicative with parameters* $\zeta_{q,r}$. The operator mapping L_n to L_{n+1} has two important, related properties. First, it is commutative: the order of the successive items of the assessment has no effect on the result, that is, on the final probability distribution on the set of states. Second, the operator is essentially Bayesian (this was proved by Mathieu Koppen[21]).

It can be shown that, under the above Rules [Q], [R] and [U], the latent state is recoverable in the sense that:

$$\mathbf{L}_n(K_0) \xrightarrow{\text{a.s.}} 1. \tag{25}$$

We recall that "a.s." stands for "almost surely". For a proof, see Theorem 1.6.7 in Falmagne and Doignon (2011). Note that this theorem assumes that the careless errors and the lucky guesses have probability zero (the straight case as in Definition 35).

While Rules [U], [Q], and [R] are fundamental to the assessment, a straightforward implementation of these rules in an assessment engine is not feasible for two reasons. One is that students often make careless errors. A solution is to amend Rule [R] by introducing a 'careless error parameter'. This would result, for example, in the modified rule:

[21] Personnal communication; see Subsection 13.4.5 in Falmagne and Doignon (2011).

[R'] **Modified Response Rule.** For all positive integers n,

$$\mathbb{P}(\mathbf{R}_n = \iota_{K_0}(q) \bigm| \mathbf{Q}_n = q, \mathbf{L}_n, \mathbf{W}_{n-1}) = \begin{cases} 1 - \beta_q & \text{if } q \in K_0 \\ 0 & \text{if } q \notin K_0\,, \end{cases} \tag{26}$$

in which β_q is the probability of making an error in responding to the item q which lies in the latent state K_0. The parameters β_q can be estimated from the data (see Subsection 12.1, page 43). In some cases, 'lucky guess' parameters may also be used, for example to deal with cases of the multiple choice paradigm. However, in real life situations, the occurrence of lucky guesses would render the result of the assessment so unreliable as to render it practically useless. Multiple choice paradigms are cheap but their results are questionable and they should be avoided. The ALEKS assessment system, which is based on the three rules [U], [Q], and [R'], rarely use multiple choice. When it does, the number of possible responses is so large that the probability of a lucky guess is negligible.

The second reason is that, in many real life applications, the domain formalizing an actual curriculum is typically very large, containing several hundred items. The learning space can still be constructed but its collection of states is huge, counting many million states. It means that the assessment algorithm cannot proceed in the straightforward manner outlined above: the probability distribution on such a large collection of states cannot be managed. We now expose a solution.

10.2 The parallel algorithm for large domains

In case the domain Q of the 'parent learning space' $\mathcal{L}$ is large, the algorithm first partitions Q into N subdomains Q^1, Q^2, ..., Q^N. Here, the subdomains are approximately of equal sizes, and they are manageable because N is chosen sufficiently large. They are representatives of the parent domain Q, in the sense, for example, that each one of them could serve as a placement test. By the Projection Theorem 33(i) (page 26), these N subdomains determine N projections $\mathcal{L}^1$, $\mathcal{L}^2$, ..., $\mathcal{L}^N$ which are all learning spaces. The algorithm then manages N simultaneous[22] assessments on the N projected learning spaces $\left(Q^i, \mathcal{L}^i\right)$ (where $i = 1, 2, \ldots, N$). The key difficulty is to immediately transfer information obtained from a response to some item q in the domain Q^j of the projected learning space $\mathcal{L}^j$, to all the other $N - 1$ learning spaces with appropriate updates of their state probabilities.

The main features of the algorithm are described in Falmagne et al. (2013, see Section 8.8, pages 143–144). The basic ideas are as follows.

1) On each trial, scan the N learning spaces and choose the "most informative" item to be presented to the student. Suppose that this item is q, belonging to the subdomain Q^i of the projected learning space $(Q^i, \mathcal{L}^i)$.

2) Record the student's response to item q. Update the state probabilities of the learning space $\mathcal{L}^i$.

[22] or maybe better said: alternating.

3) Update the state probabilities of all the other learning subspaces $\mathcal{L}^j$, $j \neq i$. This is achieved by temporarily adding item q to the other learning spaces $\mathcal{L}^j$ and working with extended distributions (cf. Definition 37).

4) At the end of the assessment, combine the information gathered in all the N learning subspaces into a final knowledge state in $\mathcal{L}$. For details, see Falmagne et al. (2013, Subsection 8.8.1, page 144).

11 About Building Knowledge Spaces or Learning Spaces

Building a knowledge space on a given domain is a highly demanding, data driven enterprise. Building a learning space is an even much more intricate process. Chapters 15 and 16 of Falmagne and Doignon (2011) describe a method in details. We only sketch the main ideas here, proceeding in two steps. We first describe some theoretical constructions which are instrumental for building knowledge spaces. We then show how these tools can be amended in the case of learning spaces.

The basic algorithm is the '`QUERY`' routine, which is due to Koppen (1993) and Müller (1989) (see also Dowling, 1994). The routine manages the following type of queries, which are either put to expert teachers or have responses inferred from assessment data.

[Q] *Suppose that a student under examination has just provided wrong responses to all the items in some set A. Is it practically certain that this student will also fail item q? Assume that the conditions are ideal in the sense that errors and lucky guesses are excluded.*

Such a *query* is summarized as the pair (A, q), with $\varnothing \neq A \subset Q$ and $q \in Q$. The `QUERY` routine gradually builds the relation $\mathcal{P}$ from 2^Q to Q consisting of all the queries (A, q) which receive a positive response. For $m = |Q|$, there are $m\left(2^{m-1} - 1\right)$ useful queries (asking a query (A, q) with $q \in A$ is useless: the answer must be affirmative). So their number is superexponential in m. Fortunately, in most cases, only a small fraction of them need to be asked: as we explain below, the responses to some queries may be inferred from the response to previously asked queries. The importance of the queries regarding the problem at hand—building a knowledge space or a learning space—lies in the following theorem (Koppen and Doignon, 1990).

Theorem 38. *For a finite domain Q, there is a one-to-one correspondence between the collection of knowledge spaces $\mathcal{K}$ on Q and the set of relations $\mathcal{P}$ from $2^Q \setminus \{\varnothing\}$ to Q satisfying for all q in Q and A, B in $2^Q \setminus \{\varnothing\}$:*

(i) *if $q \in A$, then $A\mathcal{P}q$;*

(ii) *if $A\mathcal{P}b$ for all b in B and $B\mathcal{P}q$, then $A\mathcal{P}q$.*

One such correspondence is defined by the two equivalences, where $r \in Q$, K, $B \subseteq Q$,

$$K \in \mathcal{K} \iff \big(\forall (A, q) \in \mathcal{P} :\ A \cap K = \varnothing \implies q \notin K\big), \tag{27}$$

$$(B, r) \in \mathcal{P} \iff \big(\forall K \in \mathcal{K} :\ B \cap K = \varnothing \implies r \notin K\big). \tag{28}$$

In view of Formula (27), a positive answer to the query (A, q) may lead to remove some subsets from the collection of potential states (namely, all the subsets K satisfying both $A \cap K = \varnothing$ and $q \in K$). Negating both sides of Formula (27) gives us the equivalent formula

$$K \notin \mathcal{K} \iff \big(\exists (A, q) \in \mathcal{P} :\ A \cap K = \varnothing \text{ and } q \in K\big), \tag{29}$$

which clearly shows that any non-state K is ruled out by a positive answer to some query. Thus Theorem 38 is instrumental for building a knowledge space—which may or may not be a learning space. We outline below (starting on page 39) how we can adapt the implementation of Theorem 38 so that learning spaces are generated.

In the QUERY procedure, Block j consists of all the queries (A, q) with $j = |A|$. The procedure starts with Block 1, followed by Block 2, etc. In principle an expert teacher is able to provide the answer to any query in Block 1, because such a query takes the following simple form.

[Q1] *Suppose that a student under examination has just provided a false response to question q. Is it practically certain that this student will also fail item r? Assume that the conditions are ideal in the sense that errors and lucky guesses are excluded.*

Collecting the responses to such queries $(\{r\}, q)$, or (r, q) for short, with r, q in Q and $r \neq q$ amounts to constructing a relation R on the set Q. But not all such queries need to be asked. Assuming that the expert is consistent, we can infer the responses of some queries from other queries. For instance if the responses to the queries (r, t) and (t, q) are positive, then the response to (r, q) should also be positive. In general, we only need to ask the expert a relatively small set of queries of the type [Q1]. Their responses yield some relation R. The *transitive closure*[23] $t(R)$ of the relation R is a quasi order on Q. By Birkhoff's Theorem 10 (page 10), the quasi order $t(R)$ uniquely defines a quasi ordinal knowledge space on Q, that is, a knowledge structure closed under union and intersection. The design of the QUERY algorithm ensures that this quasi order is in fact a partial order. Thus the quasi ordinal structure is an ordinal space $\mathcal{L}_1$, namely, a learning space closed under intersection. The learning space $\mathcal{L}_1$ contains all the actual knowledge states of the learning space under construction. However, it also typically contains a possibly very large number of false states, which are due to the closure of intersection of $\mathcal{L}^1$.

While human experts are capable of providing useful responses to queries of the type [Q1], their responses to queries of higher blocks are less reliable.

[23] That is, the smallest transitive relation on Q that includes R.

Fortunately, despite the presence of false states, the learning space $\mathcal{L}_1$ is sufficiently informative to be used in the schools and colleges[24]. The data collected by assessments using $\mathcal{L}_1$ can then be used to simulate human expert responses to queries of higher block numbers, such as Block 2 or Block 3.

Here is how, taking Block 2 as our example. Assuming that we have a very large collection of assessment data[25], we can estimate the conditional probabilities that failing both items r and t entails failing also item q. Using a convenient abuse of notation, we write for these conditional probabilities[26]:

$$\mathbb{P}(q = 0 \mid r = t = 0). \tag{30}$$

Choosing a suitably large threshold value θ, with $0 < \theta < 1$, we can construct the relation $\mathcal{P}$ for Block 2 by the rule:

$$\{r, t\}\mathcal{P}q \qquad \text{exactly when} \qquad \mathbb{P}(q = 0 \mid r = t = 0) > \theta. \tag{31}$$

Using such estimates and Formula (29) help implementing Block 2 of the QUERY routine. However, the resulting knowledge space is not necessarily a learning space. Consequently, not all positive responses $\{r, t\}\mathcal{P}q$ should be implemented and result in the elimination of states. This remark also applies in the general case, to the queries of any block.

The QUERY routine has been adapted for the construction of learning spaces. As suggested by our example of Block 2, the general idea is to verify that the removal of a state by the application of Theorem 38 would not result in a violation of the learning space axioms. The key result is Theorem 16.1.6 in Falmagne and Doignon (2011), which is restated as Theorem 43 here. It relies on the concept of a 'hanging state'.

Definition 39. A nonempty state K in a knowledge structure $(Q, \mathcal{K})$ is *hanging* if its inner fringe $K^{\mathcal{I}} = \{q \in Q \mid K \setminus \{q\} \in \mathcal{K}\}$ (cf. Definition 20) is empty. The state K is *almost hanging* if it contains more than one item, but its inner fringe consists of a single item.

Example 40. The knowledge space

$$\mathcal{L} = \{\varnothing, \{a\}, \{b\}, \{a, b\}, \{a, c\}, \{a, d\}, \{a, b, c\}, \{a, b, d\}, \{a, c, d\}, Q\},$$

depicted in Figure 40 has no hanging states, and has two almost hanging states, which are $\{a, c\}$ and $\{a, d\}$. According to Theorem 41 below, $\mathcal{L}$ must be a learning space.

The following result is straightforward.

Theorem 41. *A finite knowledge space is a learning space if and only if it has no hanging state.*

[24] The likely reason is that any false state L may be close to some true state K (in the sense of the symmetric difference distance $|K \triangle L| = |(L \setminus K) \cup (K \setminus L)|$).

[25] Which is the typical case in the ALEKS system for example.

[26] Technically, q, r, and t denote items and not random variables

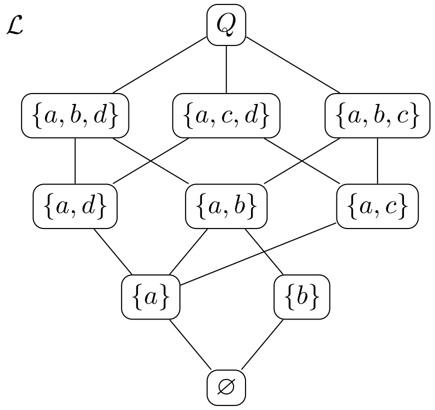

Figure 15: The covering diagram of the knowledge space in Example 40.

Theorem 41 indicates that, in the application of the QUERY routine, we must avoid the creation of hanging states (which could result from the removal of states by a positive response to a query). We need one more tool to analyze the effect of such a positive response.

Definition 42. Let $(Q, \mathcal{K})$ be a knowledge space and let (A, q) be any query such that $\varnothing \neq A \subset Q$ and $q \in Q \setminus A$. For any subfamily $\mathcal{F}$ of $\mathcal{K}$, we define

$$\mathcal{D}_{\mathcal{F}}(A, q) = \{K \in \mathcal{F} \bigm| A \cap K = \varnothing \text{ and } q \in K\}. \tag{32}$$

Thus, $\mathcal{D}_{\mathcal{K}}(A, q)$ is the subfamily of all those states of $\mathcal{K}$ that would be removed by a positive response $A\mathcal{P}q$ to the query (A, q) in the framework of the QUERY routine.

Theorem 43. *For any knowledge space $\mathcal{K}$ and any query (A, q), the family of sets $\mathcal{K} \setminus \mathcal{D}_{\mathcal{K}}(A, q)$ is a knowledge space. If $\mathcal{K}$ is a learning space, then $\mathcal{K} \setminus \mathcal{D}_{\mathcal{K}}(A, q)$ is a learning space if and only if there is no almost hanging state L in $\mathcal{K}$ such that $A \cap L = L^{\mathcal{I}}$ and $q \in L$.*

The *adapted* QUERY routine for building a learning space is based on Theorem 43. As we described above (page 38), we obtain a learning space (actually an ordinal space) $\mathcal{L}_1$ at the end of Block 1. The next query from Block 2 is of the form $(\{r, t\}, q)$. A positive response $\{r, t\}\mathcal{P}q$ induced from the estimated conditional probabilities of Statement (31) would lead to the removal from $\mathcal{L}_1$ of any state K such that

$$\{r, t\} \cap K \;=\; \varnothing \quad \text{and} \quad q \in K, \tag{33}$$

provided $\mathcal{L}_1 \setminus \{K\}$ does not contain any hanging state (cf. Formula (29)). But there is one more subtlety. The fact that $\mathcal{L}_1 \setminus \{K\}$ contains some hanging state does not necessary mean that the query $(\{r, t\}, q)$ must be discarded for ever. Indeed, it may be that, at a later stage of the execution of the algorithm, after the removal of some other states, implementing $\{r, t\}\mathcal{P}q$ no longer creates hanging states. The adapted QUERY routine takes care of such details (see Falmagne and Doignon, 2011, Chapter 15 and 16).

So far, only Block 2 of the adapted QUERY routine has been implemented to built the learning space $\mathcal{L}_2$ in realistic situations involving several hundred items, such as Beginning Algebra as the subject is taught in the US. Remarkably, in these cases, it was observed that 99% of the false states in $\mathcal{L}_1$ are removed to form $\mathcal{L}_2$. It is highly plausible that even better, smaller learning spaces would result from implementing Block 3 to $\mathcal{L}_2$, and maybe even higher blocks later on. We have been told by the ALEKS team[27] that this costly enterprise is still at the project stage.

Doignon (2014) recently proposed another modification of the QUERY routine to build a learning space. He uses a fundamental property of the collection of all the knowledge structures on the domain Q, a property which follows from results in Caspard and Monjardet (2004). Here, a knowledge structure $\mathcal{K}$ on the domain Q is included in a knowledge structure $\mathcal{L}$ also on Q when any state in $\mathcal{K}$ is also a state in $\mathcal{L}$ (that is, $\mathcal{K} \subseteq \mathcal{L}$).

Theorem 44. *For a given finite knowledge space $(Q, \mathcal{K})$, there are two cases. Either there is no learning space on Q included in $\mathcal{K}$, or among all the learning spaces on Q included in $\mathcal{K}$, there is one which includes all the other ones.*

When the second case in Theorem 44 occurs, we denote by $\mathcal{K}^{\triangle}$ the largest learning space included in the knowledge space $\mathcal{K}$. Theorem 46 below provides a description of $\mathcal{K}^{\triangle}$ in terms of 'gradations' (Theorems 44 and 46 first appeared in Doignon, 2014).

Definition 45. Let $(Q, \mathcal{K})$ be a finite knowledge structure. A *learning path* in $\mathcal{K}$ is a maximal chain of states. A *gradation* $\mathcal{C}$ is a learning path which is accessible (Definition 5), that is: for any nonempty state K in $\mathcal{C}$ there is some item q in K for which $K \setminus \{q\} \in \mathcal{K}$.

Notice that a learning path necessarily contains both states $\varnothing$ and Q. A gradation always consists of $1 + |Q|$ states of respective sizes $0, 1, \ldots, |Q|$, and it forms itself a learning space on Q. There is an obvious correspondence between gradations and learning strings (Definition 23).

Theorem 46. *Let $\mathcal{K}$ be a knowledge space on the finite domain Q. If $\mathcal{K}$ includes at least one learning space on Q, then the largest learning space $\mathcal{K}^{\triangle}$ on Q included in $\mathcal{K}$ is formed by all the states in all gradations in $\mathcal{K}$.*

As the classical QUERY routine, the *adjusted QUERY routine* proposed in Doignon (2014) repeatedly asks queries and maintains a collection $\mathcal{L}$ of subsets of Q which, although it is decreasing, always forms a learning space on Q. When the query (A, q) prompts a positive answer from the expert (or the

[27] Personnal communication.

database), the routine builds the collection $\mathcal{K} = \mathcal{L} \setminus \mathcal{D}_{\mathcal{L}}(A, q)$. By Theorem 43, the collection $\mathcal{K}$ is again a knowledge space. Now there are two cases:

(i) $\mathcal{K}$ includes some learning space (in other words: there is at least one gradation in $\mathcal{K}$, cf. Theorem 46). Then, in view of Theorem 44, $\mathcal{K}$ includes for sure the learning space $\mathcal{K}^{\triangle}$. The routine then replaces $\mathcal{L}$ with the latter learning space and asks a new query, if any is left unanswered.

(ii) $\mathcal{K}$ does not include any gradation. Then the routine exits execution, delivering the actual collection $\mathcal{L}$.

Killing the routine in Case (ii) above makes sense, because there is no learning space that complies with the answers collected to queries. To the contrary, when the query answers are coherent in the sense that they reflect some latent learning space, Case (ii) does not occur. Even more, the routine uncovers the latent learning space.

Theorem 47. *If $\mathcal{L}$ is a latent learning space on the finite domain Q and the query answers are truthful with respect to $\mathcal{L}$, then the adjusted* `QUERY` *routine will ultimately uncover $\mathcal{L}$.*

In any case, the adjusted `QUERY` routine always produces a learning space (even in case it reached Case (ii) above and therefore exited execution). We will not go into the details of the implementation of the adjusted `QUERY` routine, nor report any comparison of its performances with those of the adapted `QUERY` routine.

12 Some Applications—The ALEKS System

In psychometrics, the concepts of "validity" and "reliability" are distinct[28] and dealt with separately, which is justified because a psychometric test is a measurement instrument which is not automatically predictive of a criterion. In the applications of Learning Space Theory (LST) such as ALEKS, however, the items potentially[29] used in any assessment are, by construction, a fully comprehensive coverage of a curriculum, typically based on the consultation of numerous standard textbooks. This implies that if a LST-type assessment is reliable, it is presumably also valid, and vice versa (cf. Section 17.1 in Falmagne and Doignon, 2011).

In our introduction, we have presented LST as a more predictive, or valid, alternative to psychometrics. We describe here four analysis of data which all vindicate that position. The material is taken from Cosyn et al. (2013). However, the statistical analysis is so complex and detailed that we can only provide a summary here.

[28] See Anastasi and Urbina (1997) for explanations.

[29] Potentially, in the sense that any item of the domain could be part of the test.

12.1 The extra problem method

The first inquiry investigates whether the knowledge state uncovered at the end of an assessment is predictive of the responses to questions that were not asked. For example, there are 416 items in the elementary school mathematics domain used in the ALEKS system, of which at most 35 are used in any assessment. Can we reliably predict the student's responses to the $381 = 416 - 35$ remaining questions?

To find out, an *extra problem* is asked in any assessment, the response to which is not used in uncovering the state. The authors of Cosyn et al. (2013) have evaluated the correlation between the response to the extra problem—coded as 1/0 for correct/false—and a couple of predictive indices obtained from or during the assessment.

One predictive index is the dichotomic variable coding the observation that the extra problem is either in or out of the final state. Accordingly, the data takes the form of the 2 by 2 matrix in Table 5.

Table 5: Basic data matrix for the computation of the correlation between the cases "in or out of the final state" and the student's response coded as 1/0 for correct/false. So, theoretically, z stands for the number of lucky guesses, and y for the number of careless errors.

		Response	
		1(correct)	0 (false)
State	in	x	y
	out	z	w

Because no correlation coefficient is available that would be fully adequate for the situation, two potentially useful ones were computed by Cosyn and his colleagues for these data: the tetrachoric and the phi coefficient. The results are given in Table 6 for the median values of the coefficients and the grouped data. These data pertain to 125,786 assessments using 324 problems out of the 370 problems mentioned above[30]. The tetrachoric values are in the left column of the table. The median of the distribution is around .68. The grouped data, obtained from gathering the 324 individual 2×2 matrices into one, yields a higher correlation of about .80. These are high values, but the tetrachoric coefficient is regarded as biased upward (however, see Greer et al., 2003). The right column of the table contains similar values for the phi coefficient. These values are much lower, yielding a median of .43 (contrasting with the .68 value obtained for the tetrachoric) and a grouped data value of .58 (instead of .80).

[30] Forty-six problems were discarded because the relevant data were not sufficient to provide reliable estimates of the coefficients.

Table 6: Values of the coefficients correlating the 1/0 variables coding the in/out of the final state and the correct/false response in the two cases: median value of the correlation distribution and grouped data.

	tetrachoric	phi
median	.68	.43
grouped data	.80	.58

12.2 Correcting for careless errors

The relatively low correlation values obtained for the phi coefficient in the above analysis may be due in part to the occurrence of careless errors. However, the basic data matrix of Table 6 may be revised to include such careless errors. Instead of the dichotomic *in/out (of the final state)* variable, Cosyn et al. (2013) define the new variable

$$\mathbf{S_a} = \begin{cases} 1 - \epsilon_{\mathbf{a}} & \text{if the final state contains the extra question } \mathbf{a}, \\ 0 & \text{otherwise,} \end{cases} \tag{34}$$

in which $\epsilon_{\mathbf{a}}$ stands for the probability of committing a careless error to item **a**. So, if the extra question **a** is contained in the uncovered state, $\mathbf{S_a}$ is the probability of not committing a careless error to item **a**. The careless errors probabilities $\epsilon_{\mathbf{a}}$ were estimated from those cases in which, by chance, the same item **a** appears twice in an assessment, once as the extra problem and the other as one of the other items, and moreover, the response in at least one of the two instances of **a** is correct. The relative frequency of cases in which a false response is given to the other instance of **a** provides an estimate of $\epsilon_{\mathbf{a}}$.

The variable $\mathbf{S_a}$ is neither exactly continuous nor exactly discrete[31]. Nevertheless, for the purpose of comparison with similar analyses performed in psychometric situations, the authors have used the point biserial coefficient r_{pbis} to compute the correlation between the variables $\mathbf{S_a}$ and the 1/0 variable coding the correct/incorrect responses to the extra problem. The value reported for r_{pbis} was .67, noticeably higher than the .58 obtained for the phi coefficient for the same grouped data.

The authors compare this .67 value for the point biserial coefficient with those disclosed for the same coefficient in the Educational Testing Services (ETS) (2008) report[32] for the Algebra I California Standards Test (CST), which covers approximately the same curriculum as the `ALEKS` assessment for elementary algebra and is given to more than 100 000 students each year. This test consists

[31] For example, the distribution of $\mathbf{S_a}$ vanishes in a positive neighborhood of 0, but is positive at the point 0 itself.

[32] Produced for the California Department of Education (Test and Assessment Division). See `http://www.cde.ca.gov/ta/tg/sr/documents/csttechrpt07.pdf`.

of 65 multiple choice questions (items) and is built and scored in the framework of Item Response Theory (IRT), for which the point biserial coefficient is a standard measure. In particular, for each of the 65 items, a point biserial item-test correlation was computed, which measured the relationship between the dichotomous variable giving the 1/0 item score (correct/incorrect) and the continuous variable giving the total test score (see p. 397 of the ETS report referenced above). For the 2007 administration of the Algebra I CST, the mean point biserial coefficient for the 65 items was .36, and the median was .38 (see Table 7.2, p. 397 of the ETS report). The minimum coefficient obtained for an item was .10 and the maximum was .53 (Table 7.A.4, pp. 408–9, of the ETS report). The averages for preceding years on the test were similar, namely, the mean point biserial coefficients were .38 in 2005 and .36 in 2006 (see Table 10.B.3, p. 553, of the same report)[33].

12.3 Learning readiness

A student using the ALEKS system is given regular assessments. At the end of each assessment[34], the system is giving the student the choice of the next item to learn, such items being located in the outer fringe of the student's knowledge state[35]. This makes sense because the items in the outer fringe are exactly those that the student is ready to learn. So, if the validity of the assessment is high, the probability of successfully learning an item chosen in the outer fringe should also be high. Cosyn et al. (2013) have estimated the probability that a student successfully learns an item chosen in the outer fringe of his or her state. For elementary school mathematics, the median of the distribution of the estimated (conditional) success probabilities was .93. This estimated was based on 1,940,473 learning occasions.

12.4 ALEKS based placement at the University of Illinois

Until 2006, the students entering the University of Illinois had to take a mathematics placement test based on the ACT[36]. The results were not satisfactory because many students lack an adequate preparation for the course they were advised to take and ended up withdrawing. Beginning in 2007, the ACT was replaced by ALEKS. Ahlgren and Harper (2013) reports a comparison of the two situations, which we briefly sum up here (see also Ahlgren and Harper, 2011).

The consequence of a withdrawal from a course may have dire consequences. For example, we read in the introductory section of Ahlgren and Harper (2013):

> *"At the University of Illinois, the standard introductory calculus course (Calc: Calculus I) is a five credit course, the withdrawal from*

[33] These low correlation values were obtained even though some items with low item test correlations were removed from the test in a preliminary analysis.

[34] Except the final one, at the end of the course.

[35] See Section 6 for the concept of outer fringe.

[36] Originally, an acronym for *American College Testing*.

Table 7: Percentages of decrease in withdrawals from three courses between Fall 2006 (before ALEKS) and Fall 2007, Fall 2008.

	Percentage of Decrease in Withdrawals		
Course	**BusCalc**	**Calc**	**CalcExp**
2006–2007	↘57%	↘49%	↘67%
2006–2008	↘19%	↘81%	↘42%

> *which beyond the add-deadline may reduce students to a credit total below full-time status, resulting in the loss of tuition benefits, health benefits, scholarships, and athletic eligibility.*"

The placement program of the Department of Mathematics at the University of Illinois deal with four courses: Preparation for Calculus (**Pre-Calc**), Calculus 1 (**Calc**), Calculus 1 for student with experience (**CalcExp**), and Business Calculus (**BusCalc**). Because the **Pre-Calc** course was only offered for the first time in 2007, it is not included in the comparison statistics below.

The new placement exam is an ALEKS assessment focused on the module[37] 'Prep for Calculus', with some items removed that are not part of the relevant course at the University of Illinois. The students take the non-proctored assessment either at home or on campus. Students failing to reach the required score for placement in one of the four courses have the options of taking the placement test again or using the ALEKS learning module (or some other method) to improve their results. The students can keep retaking assessments until the course add-deadline.

Table 7 on page 46 gives the percentages of the decrease in the numbers of withdrawals observed in 2007 and 2008 for the **BusCalc**, **Calc** and **CalcExp** courses, in comparison with the numbers of withdrawals from the same courses in 2006.

The decrease in the percentages of withdrawals is remarkable. The author also note a shift in the enrollment—not reported here—from the least advanced course, **BusCalc**, to the more advanced one, which is **CalcExp**. Presumably, this may be due to the combination of two factors: the accuracy of the assessment in the ALEKS system, which may improve the placement, and the fact the students were given the possibility of using the system to bridge some gaps in their expertise.

[37] A particular learning space used in the ALEKS system.

13 Bibliographical Notes

The first paper on knowledge spaces was Doignon and Falmagne (1985), which contains some of the combinatorial basis of the theory, and in particular one of two key properties, namely, the closure under union: a knowledge space is a knowledge structure closed under union. Doignon and Falmagne (1988) was a technical follow up. The stochastic part of the theory was presented in the two papers by Falmagne and Doignon (1988a,b). The second paper also introduces a second key property, wellgradedness. A comprehensive, non-technical description of these basic ideas is contained in Falmagne et al. (1990). (See also Doignon and Falmagne, 1987; Falmagne, 1989; Doignon, 1994, for other introductory papers.)

Even though, taken together, the closure under union and wellgradedness form a legitimate basis of the theory, it is not obvious that these axioms are pedagogically sound. Around 2002, Falmagne proposed a reaxiomatization in the form of the conditions labelled here as [L1] and [L2] (see Subsection 3 on page 5), which he called a *learning space*. These axioms were unpublished at the time, but they were communicated to Eric Cosyn and Hasan Uzun, who then proved that a knowledge structure is a learning space if and only if it is a well-graded knowledge space (see Cosyn and Uzun, 2009, and our Theorem 7 on page 8). A similar result is contained in Korte et al. (1991). In fact, learning spaces are exactly the same structures as antimatroids or convex geometries. The origin of the term antimatroid (in the setting of intersection-closed families of subsets) goes back to Jamison (1980) and Jamison (1982); see also Edelman (1986) for a lattice-theoretic approach, and Edelman and Jamison (1985).

In time, other scientists became interested in knowledge space theory, notably Cornelia Dowling[38], Jürgen Heller and Reinhard Suck in Germany, Dietrich Albert and Cord Hockemeyer in Austria, Mathieu Koppen in The Netherlands, and Luca Stefanutti in Italy. The literature on the subject grew and now contains several hundred titles. The data base at
`http://liinwww.ira.uka.de/bibliography/Ai/knowledge.spaces.html`
contains an extensive list of reference on knowledge spaces. It is maintained by Cord Hockemeyer at the University of Graz (see also Hockemeyer, 2001).

The monograph by Doignon and Falmagne (1999) gives a comprehensive description of knowledge space theory at that time. Beginning in 1994, an internet-based educational software based on learning space theory was developed at the University of California, Irvine (UCI), by a team of software engineers including (among many others) Eric Cosyn, Damien Lauly, David Lemoine and Nicolas Thiéry, under the supervision of Falmagne. This was supported by a large NSF grant to UCI. The end product of this software development was called `ALEKS`, which is also the name of a company created in 1996 by Falmagne and his graduate students[39]. Around 1999, it was thought that the educational software had matured enough for a fruitful application in the schools and uni-

[38] Formerly Cornelia Müller; see this name also for references.

[39] In particular Cosyn, Lauly and Thiéry. `ALEKS` Corporation was sold to McGraw-Hill Education in 2013.

versities. This led to further developments in the form of statistical analysis of the assessment data and new mathematical results. Falmagne et al. (2006) is a not-too-technical introduction to the topic at that time. The monograph by Falmagne and Doignon (2011, a much expanded reedition of Doignon and Falmagne, 1999) is a comprehensive and (almost) up-to-date technical presentation of the theory.

Several assessment systems are founded on Knowledge Space Theory and Learning Space Theory, the most prominent ones being `ALEKS` and `RATH`. (The `RATH` system was developed by a team of researchers at the University of Graz, in Austria; see Hockemeyer, 1997 and Hockemeyer et al., 1998).

References

Ahlgren, A., and Harper, M. 2011. Assessment and Placement through Calculus I at the University of Illinois. *Notices of the American Mathematical Society*, **58**, 1460–1461.

Ahlgren, A., and Harper, M. 2013. ALEKS-based Placement at the University of Illinois. Pages 51–68 of: Falmagne, J.-Cl., Albert, D., Doble, C.W., Eppstein, D., and Hu, X. (eds), *Knowledge Spaces: Applications in Education.* Berlin: Springer.

Anastasi, A., and Urbina, S. 1997. *Psychological Testing (seventh edition).* Upper Saddle River, NJ: Prentice Hall.

Birkhoff, G. 1937. Rings of sets. *Duke Mathematical Journal*, **3**, 443–454.

Boyd, E.A., and Faigle, U. 1990. An algorithmic characterization of antimatroids. *Discrete Applied Mathematics*, **28**, 197–205.

Caspard, N., and Monjardet, B. 2004. Some lattices of closure systems on a finite set. *Discrete Mathematics and Theoretical Computer Science*, **6**, 163–190 (electronic).

Cosyn, E., and Uzun, H.B. 2009. Note on two necessary and sufficient axioms for a well-graded knowledge space. *Journal of Mathematical Psychology*, **53**, 40–42.

Cosyn, E., Doble, C.W., Falmagne, J.-Cl., Lenoble, A., Thiéry, N., and Uzun, H. 2013. Assessing mathematical knowledge in a learning space. Pages 27–50 of: Falmagne, J.-Cl., Albert, D., Doble, C.W., Eppstein, D., and Hu, X. (eds), *Knowledge Spaces: Applications in Education.* Berlin: Springer.

Doble, C.W., Doignon, J.-P., Falmagne, J.-Cl., and Fishburn, P.C. 2001. Almost connected orders. *Order*, **18**, 295–311.

Doignon, J.-P. 1994. Probabilistic assessment of knowledge. Pages 1–56 of: Albert, D. (ed), *Knowledge Structures.* New York: Springer Verlag.

Doignon, J.-P. 2014. Learning spaces, and how to build them. Pages 1–14 of: Glodeanu, C.V., Kaytoue, M., and Sacarea, Chr. (eds), *Formal Concept Analysis, 12th International Conference, ICFCA 2014, Cluj-Napoca, Romania, June 10-13, 2014.* Lecture Notes in Artificial Intelligence, vol. 8478. Berlin, Heidelberg, and New York: Springer-Verlag.

Doignon, J.-P., and Falmagne, J.-Cl. 1985. Spaces for the assessment of knowledge. *International Journal of Man-Machine Studies*, **23**, 175 – 196.

Doignon, J.-P., and Falmagne, J.-Cl. 1987. Knowledge assessment: A set theoretical framework. Pages 129–140 of: Ganter, B., Wille, R., and Wolfe, K.E. (eds), *Beiträge zur Begriffsanalyse: Vorträge der Arbeitstagung Begriffsanalyse, Darmstadt 1986.* Mannheim: BI Wissenschaftsverlag.

Doignon, J.-P., and Falmagne, J.-Cl. 1988. Parametrization of knowledge structures. *Discrete Applied Mathematics*, **21**, 87–100.

Doignon, J.-P., and Falmagne, J.-Cl. 1999. *Knowledge Spaces.* Berlin: Springer-Verlag.

Dowling, C. 1993. On the irredundant construction of knowledge spaces. *Journal of Mathematical Psychology*, **37**, 49–62.

Dowling, C.E. 1994. Integrating different knowledge spaces. Pages 149–158 of: Fischer, G.H., and Laming, D. (eds), *Contributions to Mathematical Psychology, Psychometrics, and Methodology.* Berlin, Heidelberg, and New York: Springer-Verlag.

Edelman, P.H. 1986. Abstract convexity and meet-distributive lattices. Pages 127–150 of: *Combinatorics and ordered sets (Arcata, Calif., 1985).* Contemp. Math., vol. 57. Providence, RI: Amer. Math. Soc.

Edelman, P.H., and Jamison, R.E. 1985. The theory of convex geometries. *Geometriae Dedicata*, **19**, 247–270.

Educational Testing Services (ETS). 2008. *California Standard Tests (CSTs) Technical Report, Spring 2007 Administration.* Tech. rept. California Department of Education—Standard and Assessment Division. Contract 5417, `http://www.cde.ca.gov/ta/tg/sr/documents/csttechrpt07.pdf`.

Eppstein, D. 2013a. Learning sequences: An efficient data structure for learning spaces. Pages 287–304 of: Falmagne, J.-Cl., Albert, D., Doble, C.W., Eppstein, D., and Hu, X. (eds), *Knowledge Spaces: Applications in Education.* Berlin: Springer.

Eppstein, D. 2013b. Projection, decomposition, and adaption of learning spaces. Pages 305–322 of: Falmagne, J.-Cl., Albert, D., Doble, C.W., Eppstein, D., and Hu, X. (eds), *Knowledge Spaces: Applications in Education.* Berlin: Springer.

Falmagne, J.-C., and Doignon, J.-P. 2011. *Learning Spaces.* Berlin: Springer-Verlag.

Falmagne, J.-C., Albert, D., Doble, C.W, Eppstein, D., and Hu, X. (eds). 2013. *Knowledge Spaces: Applications in Education.* Berlin: Springer-Verlag.

Falmagne, J.-Cl. 1989. Probabilistic knowledge spaces: A review. In: Roberts, Fred (ed), *Applications of Combinatorics and Graph Theory to the Biological and Social Sciences, IMA Volume 17.* New York: Springer Verlag.

Falmagne, J.-Cl., and Doignon, J-P. 1988a. A class of stochastic procedures for the assessment of knowledge. *British Journal of Mathematical and Statistical Psychology*, **41**, 1–23.

Falmagne, J.-Cl., and Doignon, J-P. 1988b. A Markovian procedure for assessing the state of a system. *Journal of Mathematical Psychology*, **32**, 232–258.

Falmagne, J.-Cl., Koppen, M., Villano, M., Doignon, J.-P., and Johanessen, L. 1990. Introduction to knowledge spaces: how to build, test and search them. *Psychological Review*, **97**, 204–224.

Falmagne, J.-Cl., Cosyn, E., Doignon, J.-P., and Thiéry, N. 2006. The assessment of knowledge, in theory and in practice. Pages 61–79 of: Ganter, B., and Kwuida, L. (eds), *Formal Concept Analysis, 4th International Conference, ICFCA 2006, Dresden, Germany, February 13–17, 2006.* Lecture Notes in Artificial Intelligence. Berlin, Heidelberg, and New York: Springer-Verlag.

Ganter, B., and Reuter, K. 1991. Finding All Closed Sets: A General Approach. *Order*, **8**, 283–290.

Greer, T., Dunlap, W.P., and Beaty, G.O. 2003. Intensity discrimination with gated and continuous sinusoids. *Educational and Psychological Measurement*, **63**, 931–950.

Hockemeyer, C. 1997. *RATH - a relational adaptive tutoring hypertext WWW–environment.* Technical report 1997/3. Institut für Psychologie, Karl-Franzens-Universität Graz, Austria. Online available at http://css.uni-graz.at/rath/publications/documentation.pdf.

Hockemeyer, C. 2001. *KST Tools User Manual.* Unpublished Technical Report. Institut für Psychologie, Karl-Franzens-Universität Graz, Austria. Available: http://wundt.uni-graz.at/kst.html.

Hockemeyer, C., Held, Th., and Albert, D. 1998. RATH—A relational adaptive tutoring hypertext WWW–environment based on knowledge space theory. Pages 417–423 of: Alvegard, Chr. (ed), *CALISCE‘98: Proceedings of the Fourth International Conference on Computer Aided Learning in Science and Engineering.* Göteborg, Sweden: Chalmers University of Technology.

Jamison, R.E. 1980. Copoints in antimatroids. Pages 535–544 of: *Proceedings of the Eleventh Southeastern Conference on Combinatorics, Graph Theory and Computing (Florida Atlantic Univ., Boca Raton, Fla., 1980), Vol. II*, vol. 29.

Jamison, R.E. 1982. A perspective on abstract convexity: classifying alignments by varieties. Pages 113–150 of: *Convexity and Related Combinatorial Geometry (Norman, Okla., 1980).* Lecture Notes in Pure and Appl. Math., vol. 76. New York: Dekker.

Koppen, M. 1993. Extracting human expertise for constructing knowledge spaces: An algorithm. *J. Math. Psych.*, **37**, 1–20.

Koppen, M. 1998. On alternative representations for knowledge spaces. *Math. Social Sci.*, **36**, 127–143.

Koppen, M., and Doignon, J.-P. 1990. How to build a knowledge space by querying an expert. *Journal of Mathematical Psychology*, **34**, 311–331.

Korte, B., Lovász, L., and Schrader, R. 1991. *Greedoids.* Algorithms and Combinatorics, vol. 4. Berlin: Springer-Verlag.

Müller, C.E. 1989. A procedure for facilitating an expert's judgments on a set of rules. Pages 157–170 of: Roskam, E.E. (ed), *Mathematical Psychology in Progress.* Recent Research in Psychology. Berlin, Heidelberg, and New York: Springer-Verlag.