Knowledge Structures and Didactic Model Selection in Learning Object Navigation

Dietrich Albert, Luca Stefanutti

Knowledge Structures and Didactic Model Selection in Learning Object Navigation

Dietrich Albert
Cognitive Science Section, Department of Psychology
University of Graz, Austria
dietrich.albert@uni-graz.at

Luca Stefanutti
Cognitive Science Section, Department of Psychology
University of Graz, Austria
luca.stefanutti@uni-graz.at

Abstract

In this paper two levels of content aggregation are considered: (i) a 'conceptual level' of the content of the learning objects, and (2) a 'didactic preference level' of the specific preferences expressed by a taecher or educational institution. We then present a knowledge structure model in which the learning objects are organized according to constraints imposed at these two distinct levels. The structure on the learning objects is then used to restrict the number of learning paths available to a learner during her/his navigation of the learning environment. In particular, only paths that are consistent with both the content constraints and the didactic constraints are available.

1 Introduction

According to the SCORM (Shareable Content Object Reference Model) specification, an e-learning system is broken into two main and independent components of learning content interoperability: the so called Learning Mangement Systems (LMS), and the Shareable Content Objects (SCOs). In this model the learning objects (the SCOs) are assumed to be standardized and reusable units. The process of finding and organizing the learning objects into a specific structure is called *content aggregation*. This process is responsible, for instance, of deciding which learning object comes next in the learner's interaction with the LMS. In doing so, the system may take into account a number of different aspects related to the specific content of the learning objects, the didactic preferences of the teacher or educational institution which applies the system, and the learner's personal preferences.

Elsewere some methods and models were presented for structuring reusable distributed learning objects according to their content (Stefanutti, Albert & Hockemeyer, in press). These models are based on a mathematical framework called knowledge space theory (Albert & Lukas, 1998; Doignon & Falmagne, 1999), whose key concepts and applications are summarized in the next section. In this paper the attention is focused on some didactic aspects that may be considered in structuring a set of learning objects, as well as on some ways of integrating the didactic preferences, expressed e.g. by a teacher, with the content structure on the learning objects.

We make a distinction between two levels of content aggregation. At the first level the learning objects are organized and related one another on the basis of their content in terms of the concepts and notions they contain. The structure obtained in this way puts constraints on the possible sequences of learning objects presented to the learner. At this level the teacher's didactic preferences have no influence on the structure. However, a second level of content aggregation is provided in which (at least some of the) didactic preferences of the teacher are transformed into rules that put further constraints on the structure of the learning objects. Combining together these two levels results in a learning environment in which the possible learning paths (sequences of learning objects) are consistent with both the content of the learning objects, and the preferences of the teacher.

2 Knowledge structures

Knowledge space theory (Doignon & Falmagne, 1985; Albert & Lukas, 1999; Doignon & Falmagne, 1999) offers a rigorous and efficient formal framework for the construction, validation, and application of e-assessment and e-learning adaptive systems. It is a psychometric theory for the assessment and acquisition of knowledge; however, as far as adaptivity and optimal learning paths are concerned, its methods and models can be fruitfully exploited for e-learning purposes as well.

According to this theory, a domain of knowledge is a collection Q of problems in a given field of knowledge (e.g., mathematics, physics, chemistry, biology, etc.). Then, the knowledge state of a student is the set K of all problems in Q that this student actually masters, and a knowledge structure for Q is a pair $(Q, \mathcal{K})$ in which $\mathcal{K}$ is the collection of all knowledge states that can be observed in a certain population of students. If $\mathcal{K}$ is closed under union (i.e., $K \cup K' \in \mathcal{K}$ whenever $K, K' \in \mathcal{K}$) then it is called a knowledge space. Sometimes also closure under intersection holds ($K \cap K' \in \mathcal{K}$ whenever $K, K' \in \mathcal{K}$), in which case, $\mathcal{K}$ is called a quasi-ordinal knowledge space.

The above-mentioned theory is at the basis of some existing e-learning and assessment adaptive systems in the U.S. and in Europe. Two of them are the ALEKS (Adaptive LEarning with Knowledge Spaces) system developed by the research group of Irvine, CA supervised by Jean-Claude Falmagne (http://www.aleks.com), and the RATH prototype (Relational Adaptive Tutoring Hypertext) system of the research group of Graz, Austria (http://wundt.uni-graz.at/rath, Hockemeyer et al., 1998) supervised by Dietrich Albert.

3 Learning paths and navigation history

We denote with L a set of learning objects. Each learning object in L can be a problem, an explanation, an example, and so on. All objects in L are supposed to be on the same domain of knowledge. In particular, each of them contains some specific notion or concept in this domain.

Navigating through the learning objects the learner specifies implicitly some order on the set L. The sequence in which the objects are visited is a result of the preferences and decisions of both the learner and the author of the learning management system.

In this development, with the term *learning path* we mean any sequence (linear order) of learning objects in L which is consistent with the criteria, preferences and decisions of both the learner and the author. The way in which these two actors specify their criteria and preferences, however, differs in the sense that the criteria and preferences of the author are specified in advance through some set of rules that, usually, reduce the number of possible learning paths available to the learner.

If the rules specified by the author are represented through a knowledge structure on the learning objects, i.e., by any collection $\mathcal{K}$ of subsets of L such that both $\emptyset$ and L are in $\mathcal{K}$, then, a learning path in $\mathcal{K}$ is any maximal chain $\mathcal{C}$ in the partially ordered set $(\mathcal{K}, \subseteq)$[1]. As an example consider the structure $\mathcal{K}$ on a set $L = \{1, 2, 3, 4\}$ of learning objects:

$$\mathcal{K} := \{\emptyset, \{1\}, \{2\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 3, 4\}\}.$$

Then, for instance, the collection $\mathcal{C} := \{\emptyset, \{1\}, \{1, 2\}, \{1, 2, 3\}, \{1, 2, 3, 4\}\}$ is a maximal chain in $(\mathcal{K}, \subseteq)$ and thus it is a learning path in $\mathcal{K}$. Note that this path corresponds to the sequence $(1, 2, 3, 4)$ of learning objects. On the other hand, consider the collection

$$\mathcal{C}' := \{\emptyset, \{1\}, \{1, 4\}, \{1, 2, 4\}, \{1, 2, 3, 4\}\}.$$

This is not a learning path in $\mathcal{K}$, since $\{1, 4\} \notin \mathcal{K}$. Thus the sequence $(1, 4, 2, 3)$ is not consistent with the constraints imposed by the structure $\mathcal{K}$. This means that $\mathcal{K}$ prevents the learner to navigate the learning objects in this particular sequence.

If the the learning objects visited by the learner are recorded as a set $H \subseteq L$ of learning objects, namely, the set of all learning objects already visited by her/him, then H is consistent with the constraints imposed by $\mathcal{K}$ if and only if $H \in \mathcal{K}$. In this sense, $\mathcal{K}$ contains all consistent histories.

One question that arises at this point is how an author may construct the structure $\mathcal{K}$. One very direct way would consist in specifying which of the 2^n, for $n = |L|$, subsets of L are consistent histories. This way, however, seems unfeasible for different reasons (see, in this connection, Falmagne, Koppen, Villano, Doignon, Johanessen, 1990). First, the number of such subsets grows exponentially in n and with a realistic number of learning objects, the job of separating the consistent subsets from the inconsistent ones would result in an enormous task for any human. The second reason is that there are indirect ways of establishing such collection by imposing explicit rules on the set L of the learning objects which are meaningful, for instance, from a didactic and/or a cognitive point of view. In the next section we present one possibility along this direction.

4 Concept assignments and didactic preferences

In this section we present a model that makes a distinction between two kinds of constraints on the learning history. The first kind of constraints is directly related to the content of the learning objects in terms of the concepts and the notions they contain. The second kind of constraints pertains to the different types of learning objects one can have in a learning management system. In this respect, not all learning objects are designed for the same purpose. Some of them may provide general or specific information (like definitions or explanations) on some content while others are aimed at

[1] A chain in $(\mathcal{K}, \subseteq)$ is any subset $\mathcal{C} \subseteq \mathcal{K}$ such that, for $X, Y \in \mathcal{C}$, either $X \subseteq Y$ or $Y \subseteq X$ hold. A maximal chain is a chain $\mathcal{M} \subseteq \mathcal{K}$ for which there is no other chain $\mathcal{C}' \subseteq \mathcal{K}$ such that $\mathcal{M} \subset \mathcal{C}'$.

the assessment of the knowledge of the learner (like problems or questions), still others are provided in order to give the learner concrete instances of abstract concepts and notions (like examples or exercises). The author of the learning management system might want to keep information about the content of a learning object separate from the information on its type. Keeping the content information separate from the type information the author of a learning management system might embed in the system the didactic constrainst on the content, leaving however to the final user the opportunity to choose the constraints on the types according to her/his didactic preferences.

Let C be a set of concepts (or notions) in a given field of knowledge. For example, in the field of linear algebra, a set of notions might contain the following concepts: (c_1) *vector space*, (c_2) *spanning set of vectors*, (c_3) *basis of a vector space*, (c_4) *dimension of a vector space*, (c_5) *rank of a matrix* (c_6) *linear independence*. On the other hand, we consider a set T of learning object (LO) types (or classes). Examples of learning object types are (t_1) *the LO is a problem or question*, (t_2) *the LO is an explanation or instruction* (t_3) *the LO is an example*, (t_4) *the LO is an exercise*. The pair $(T, \leqslant_T)$, where '$\leqslant_T$' is a partial order on the types in T is called a *didactic preference*. Thus, a didactic preference is expressed by (partially) ordering the different types of learning objects so that, for instance, in a 'task oriented didactic' one might require that an explanation page on a certain topic should be presented to the learner only if some problem on the same topic was already presented. In this case, if we denote with p the class of learning objects that are problems, and with e the class of learning objects that are explanations ($p, e \in T$), then we will have $p \leqslant_T e$. In general, for $t', t'' \in T$, the interpretation of $t' \leqslant_T t''$ is that a learning object belonging to class t' should be presented before (or immediately before - see Section 6 in this connection) a leaning object of class t''. We thus call the partial order $\leqslant_T$ a *predecessor relation* for the didactic preference $(T, \leqslant_T)$.

It remains to establish how concepts and types are assigned to learning objects. To this aim we introduce two distinct mappings. A *concept function* for L and C is a mapping $f : L \to 2^C \setminus \{\emptyset\}$ assigning a given subset of concepts to each learning object $x \in L$, so that $f(x) \subseteq C$ (note that $f(x) \neq \emptyset$ for all $x \in L$, that is, each learning object must have at least one concept assigned to it). On the other hand, we call *type function* a mapping $t : L \to T$ assigning a given type in T to each of the learning objects. Thus, for $x \in L$, $t(x) \in T$ is the type assigned to x.

It should be noted that the notion of a concept function is germane to that of a skill function (or skill map) introduced and discussed elsewere (see e.g., Doignon, 1994; Korossy, 1997; Doignon and Falmagne, 1999; Düntsch and Gediga, 1996). In this context, however, we prefer to use the term 'concept' rather than 'skill' since this last is closely related to cognitive aspects that we do not consider explicitly here. Instead, our focus is on the content of a learning object in a didactic sense, that is, in terms of the 'concepts' and 'notions' that can be learned through the learning object itself. Nonetheless, the mathematical concept behind the two terms 'skill function' and 'concept function' is exactly the same.

Considering any learning object in L, a distinction should be made between the concepts that a learner needs to know in order to master the learning object itself (i.e., to solve it if it is a problem or an exercise, or to understand it if it is some explanation or instruction) and those that are new (just presented in this learning object for the first time). This distinction can be found, e.g. in Hockemeyer (2002). Here we focus on the first kind of concepts and we assume that, for $x \in L$, $f(x)$ is the set of concepts required to master x. It seems thus reasonable to assume that a learning object x is a ‘prerequisite’ for some other learning object y whenever the set of concepts required by x is a subset of that required by y. Formally, we call a *prerequisite relation* for the learning objects in L a partial order $\leqslant_L$ such that, for $x, y \in L$,

$$x \leqslant_L y \quad \Longleftrightarrow \quad f(x) \subseteq f(y).$$

A partial order $\leqslant$ on the learning objects which is consistent with both the prerequisite relation $\leqslant_L$ and the didactic preference $\leqslant_T$ is thus defined by

$$x \leqslant y \quad \Longleftrightarrow \quad f(x) \subseteq f(y) \quad \text{and} \quad t(x) \leqslant_T t(y). \tag{4.1}$$

Since each learning object $x \in L$ is represented by the corresponding pair $(f(x), t(x))$, the order ‘$\leqslant$’ follows from a coordinatewise ordering of such pairs. In particular, the partial order $(L, \leqslant)$ is embedded into the Cartesian product $(\mathcal{L}, \subseteq) \times (T, \leqslant_T)$, where

$$\mathcal{L} := \{f(x) : x \in L\}.$$

This fact is established by the following

Proposition 1. *Let $(P, \leqslant_P)$ be the Cartesian product of $(\mathcal{L}, \subseteq)$ and $(T, \leqslant_T)$ (in which ‘$\leqslant_P$’ is the coordinatewise ordering). Then there exists a total function $h : L \to P$ such that, for any $x, y \in L$,*

$$x \leqslant y \quad \Longleftrightarrow \quad h(x) \leqslant_P h(y).$$

Proof. For any $x \in L$, let $h(x) = (f(x), t(x))$. Clearly, h maps L into P i.e., $h(L) \subseteq P$. Given any $x, y \in L$, from (4.1) we have $x \leqslant y$ iff $f(x) \subseteq f(y)$ and $t(x) \leqslant_T t(y)$. Moreover, the order ‘$\leqslant_P$’ is such that, for $(A, t'), (B, t'') \in P$, $(A, t') \leqslant_P (B, t'')$ iff $A \subseteq B$ and $t' \leqslant_T t''$. Thus, $x \leqslant y$ iff $h(x) \leqslant_P h(y)$ for all $x, y \in L$. $\square$

As an example, consider the set $C = \{c_1, c_2, c_3, c_4\}$ of the first four concepts on linear algebra mentioned at the beginning of this section, and the set $T = \{t_1, t_2\}$ of the first two types. An assignment of the concepts in C and types in T to a set L of ten different learning objects (numbered from 1 to 10) is displayed in Table 1.

Table 1: An example of concept and type assignment to a set of 10 learning objects (see text).

LO	c_1	c_2	c_3	c_4	Type
1	×				t_1
2	×	×			t_1
3	×		×		t_1
4	×	×	×		t_1
5	×	×	×	×	t_1
6	×				t_2
7	×	×			t_2
8	×		×		t_2
9	×	×	×		t_2
10	×	×	×	×	t_2

A cross (×) in the cell corresponding to object i and concept j indicates that concept j is assigned to learning object i. The last column of the table (Type) shows how the two

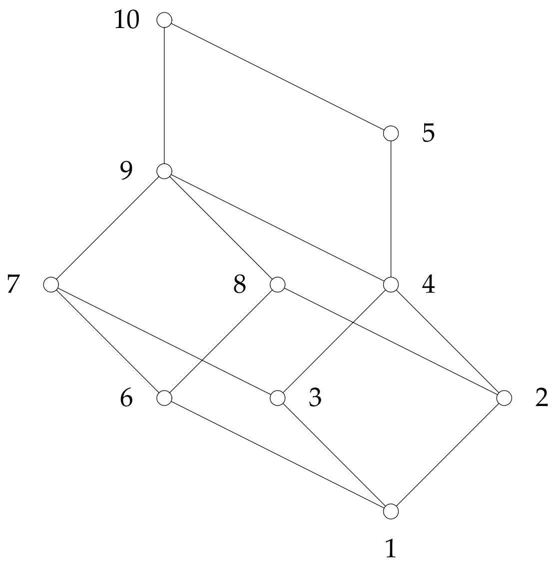

Figure 1: Partial order resulting from the application of rule (4.1) to the 10 learning objects of the example in Table 1.

types t_1 and t_2 are assigned to the learning objects. According to Table 1, for example, we have that $f(1) = \{a\}$ and $f(2) = \{a, b\}$, thus $1 \leqslant_L 2$. If now a didactic preference is expressed such that type t_1 is a predecessor of type t_2 ($t_1 \leqslant_T t_2$), the partial order resulting from an application of (4.1) is that depicted in Figure 1.

It can be observed that, if the collection $\mathcal{L} := \{f(x) : x \in L\}$ is considered:

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

then the partially ordered set $(L, \leqslant)$ is isomorphic to the cartesian product $(\mathcal{L}, \subseteq) \times (T, \leqslant_t)$ shown in Figure 2. Thus, this example represents a special case in which the

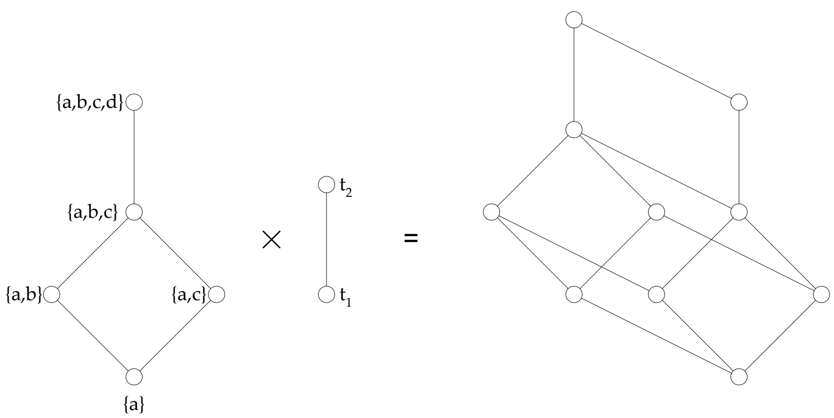

Figure 2: Cartesian product of the two partially ordered sets $(\mathcal{L}, \subseteq)$ and $(T, \leqslant_T)$.

embedding is, in fact, an isomorphism.

The question now arises of how to construct a knowledge structure $\mathcal{K}$ containing all histories $H \subseteq L$ that are consistent with the order relation '$\leqslant$' on L established by (4.1). The answer comes from a classical result in lattice theory due to Birkhoff (1967): *given a finite set X, the family of all partial orders on X is in one-to-one correspondence with the family of all collections of subsets of X that are both closed under union and under intersection*[2]. Thus, the partially ordered set $(L, \leqslant)$ corresponds to a unique collection $\mathcal{K}$ of subsets of L, and this correspondence is established by the relation

$$\mathcal{K} = \{X{\downarrow}\colon X \subseteq L\},$$

where, for $X \subseteq L$, $X{\downarrow}$ is the down-set generated by X in $(L, \leqslant)$:

$$X{\downarrow} := \{y \in L : y \leqslant x \text{ for some } x \in X\}.$$

Given a history $H \in \mathcal{K}$, its consistency with '$\leqslant$' is guaranteed by the fact that if $x \in H$ and $y \leqslant x$, then also $y \in H$ holds. Thus, for instance, the set $= \{1, 2, 6, 8\}$ is in the knowledge structure corresponding to the partially ordered set of Figure 1. The set $\{1, 2, 4, 6\}$ however, is not. In fact, 4 requires 3, which is not in this set.

5 Constrained navigation

When constraints are imposed on the learning objects in L in the form of the partial order '$\leqslant$', not all learning objects are available to a learner with history $H \subseteq L$. More precisely, which learning objects are available to her/him depends on the history H of the learner, and the partial orders $\mathcal{L}$ and T. Thus, going back to the partially ordered set of learning objects $(L, \leqslant)$ displayed in Figure 1, if for instance the history of a learner is $H = \{1, 2, 6, 8\}$, the learning object 5 should not be available to this learner simply because, moving to this object her/his new history would become $H' = \{1, 2, 5, 6, 8\}$, which is not an element of the structure $\mathcal{K}$ corresponding to $(L, \leqslant)$ (the reader can easily verify this).

In order to establish which learning objects are accessible to a learner with history H, we need some further notation. First, given two objects $x, y \in L$, we say that x is equivalent to y ($x \equiv y$) whenever both $x \leqslant y$ and $y \leqslant x$ hold true, and we denote with $[x] := \{y \in L : x \equiv y\}$ the equivalence class of x. From a didactic point of view, this is interpreted as a 'content equivalence' i.e., the two learning objects have equivalent content. Then, for $H \in \mathcal{K}$, the collection

$$H^o := \{y \in L \setminus H : H \cup [y] \in \mathcal{K}\}$$

is called the *outer fringe* of H (Doignon & Falmagne, 1999). With this new definition we require that a learning object y not in H is accessible from H whenever $y \in H^o$. In other words, if the learner's last visited learning object is $x \in H$, then a link between x and y is dynamically created for all $y \in H^o$. Thus, for $H_1 := \{1, 2, 6, 8\}$, the last visited learning object of the learner will be linked to all objects in the set $H_1^o := \{3\}$, in this case there is only one object, whose type is *problem*[3]. Note that, objects 6 and 7 are

[2] In the terminology adopted in knowledge space theory, these structures are often called *quasi-ordinal knowledge spaces* (Doignon & Falmagne, 1999).

[3] A link to all objects already visited (i.e., the elements of H), however, may always be present.

both of type *explanation* and 7 would be immediately accessible from 6 if there was no order specification on the object types. The constraint imposed on the types requires that problem 3 be visited before to access the explanations contained in 7. This just reflects the didactic preference that 'whenever compatible with the content constraints, problems should be presented before explanations'. Once problem 3 is visited, the new history will be $H_2 := \{1, 2, 3, 6, 8\}$, and the corresponding outer fringe $H_2^o := \{4, 7\}$. Table 2 shows, as an example, the learning path of a learner in the knowledge structure $\mathcal{K}$ and the corresponding set of accessible learning objects.

Table 2: Example of a learning path, the corresponding collection of outer fringes and the type of the current learning objects at each step of the learning path.

Step	History	Outer Fringe	Current Object
0	$\emptyset$	$\{1\}$	—
1	$\{1\}$	$\{2, 3, 6\}$	*problem*
2	$\{1, 2\}$	$\{3, 6\}$	*problem*
3	$\{1, 2, 6\}$	$\{3, 8\}$	*explanation*
4	$\{1, 2, 6, 8\}$	$\{3\}$	*explanation*
5	$\{1, 2, 3, 6, 8\}$	$\{4, 7\}$	*problem*
6	$\{1, 2, 3, 4, 6, 8\}$	$\{5, 7\}$	*problem*
7	$\{1, 2, 3, 4, 5, 6, 8\}$	$\{7\}$	*problem*
8	$\{1, 2, 3, 4, 5, 6, 7, 8\}$	$\{9\}$	*explanation*
9	$\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$	$\{10\}$	*explanation*
10	L	$\emptyset$	*explanation*

6 Conclusions

Adaptivity occurs at different levels. In the content aggregation process of a LMS at least two levels should be considered: (i) the level of the content of the learning objects, and (ii) that of the didactic preferences of the educational institution that applies the e-learning system. Different teachers may have different didactic preferences and the system should be able to re-organize its structure on-the-fly according to such preferences.

In this paper we presented a simple model for content aggregation based on these two levels. At the 'content' level the structure on the learning objects corresponds to partially ordering the objects according to the concepts and notions they contain. At the 'didactic preference' level learning objects are classified according to their 'purpose'. Not only learning objects differ in their content, they also have different purposes: a quiz is presented to the learner to test her/his knowledge, skills, degree of competency; an instruction is provided whenever the learner needs more information on a certain topic; an example is given to introduce to the learner some new concept or to make concrete an abstract concept, and so on. The priority of the different purposes may change in different didactic situations and educational contexts, and the LMS should be able to adapt the structure on the learning objects according to these changes.

Learning objects are thus classified according to their *types*, where different types reflect different purposes. When a partial order is specified on the learning object types, it can be combined with the order on the learning object content to produce a structure that is consistent with both the conceptual content of the objects, and the didactic preferences of the teacher. This structure can then be used to decide which learning object comes next according to content constraints, teacher's preferences, and learner's preferences.

The simple model presented in this paper can be extended in different ways. One possibility, for instance, is represented by a learning object belonging to more different types. Some learning objects can be used for different purposes, depending on the context. For instance, a problem can be used as a step-by-step exercise, or in some cases instructions may be appropriate as examples. Another possibility pertains the distinction between 'before' and 'immediately before' made in section 4. Sometimes, a learning object is required to be presented immediately after some other, like a quiz that tests the learner after some explanation. In other cases this requirement is less strict: after failing a problem, the learner might be free to choose between reading some explanation on that problem, or trying to solve some less demanding problem.

References

Albert, D., Lukas, J. (1999). *Knowledge spaces: theories, empirical research, applications.* Mahwah, NJ, London: Lawrence Erlbaum Associates.

Birkhoff, G. (1967). *Lattice theory.* Providence, R.I.: American Mathematical Society.

Doignon, J.P. (1994). Knowledge spaces and skill assignments. In G.H. Fischer and D. Laming, (Eds.), *Contributions to Mathematical Psychology, Psychometrics, and Methodology*, pp. 111-121. Springer-Verlag, New York, 1994.

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

Doignon, J.-P., Falmagne, J.-C. (1999). *Knowledge spaces.* Berlin, Heidelberg: Springer-Verlag.

Düntsch, I., Gediga, G. (1996). Skills and knowledge structures. *British Journal of Mathematical and Statistical Psychology, 48*, 9–27.

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

Hockemeyer, C., Held, T., and Albert, D. (1998). RATH — a relational adaptive tutoring hypertext WWW–environment based on knowledge space theory. In Christer Alvegard, (Ed.), *CALISCE'98: Proceedings of the Fourth International Conference on Computer Aided Learning in Science and Engineering*, pp. 417–423, Göteborg, Sweden, June 1998. Chalmers University of Technology.

Hockemeyer, C. (2002). Extending the Competence-Performance-Approach for Building Adaptive and Dynamic Tutoring Systems. Talk at the *33rd European Mathematical Psychology Group (EMPG) Meeting*, International University Bremen, Germany, August 21–24.

Korossy, K. (1997). Extending the theory of knowledge spaces: A competence performance approach. *Zeitschrift fr Psychologie, 205*, 53–82.

Stefanutti, L., Albert, D., Hockemeyer, C. (in press). Derivation of knowledge structures for distributed learning objects. *Proceedings of the 2nd International LeGE-WG Workshop on e-Learning and Grid Technologies: a fundamental challenge for Europe*, Paris, 3-4 March 2003.