Heller, J., & Repitsch, C. (2008). Distributed skill functions and the meshing of knowledge structures. Journal of Mathematical Psychology, 52, 147-157.

Available online at www.sciencedirect.com
ScienceDirect
Journal of Mathematical Psychology 52 (2008) 147–157
Journal of Mathematical Psychology
www.elsevier.com/locate/jmp

Distributed skill functions and the meshing of knowledge structures

Jürgen Heller[a,*], Claudia Repitsch[b]

[a] University of Tübingen, Germany
[b] University of Graz, Austria

Received 9 August 2006; received in revised form 11 October 2007
Available online 10 March 2008

Abstract

The scope of knowledge space theory was extended by bringing into the picture the underlying skills and capabilities that are relevant to solving the problems in a knowledge domain. A major challenge to this approach comes from the need to aggregate distributed information on (partially) overlapping domains and skill sets. The notion of a distributed skill function is introduced for formalizing the integration of several skill functions that represent the assignment of skills to problems. It is shown that their consistency is captured by the meshability of the delineated knowledge structures. This result draws upon a characterization of the meshing of finite or infinite collections of knowledge structures, which extends and generalizes previous results on the binary case. The discussion covers implications for knowledge assessment and for practical applications, such as integrating skill assignments coming from different experts or distributed resources in technology-enhanced learning.

Keywords: Knowledge space theory; Competencies; Knowledge assessment

1. Introduction

In a seminal paper Doignon and Falmagne (1985) suggested a set-theoretical framework for representing the knowledge within a certain domain, and for adaptively assessing the particular state of knowledge of an individual in a highly efficient way. From the beginning this so-called theory of knowledge spaces was developed from a behavioristic point of view. Its focus was on the solution behavior exhibited on a set of problems (questions, exercises, etc.) in a specific domain. Following Falmagne, Koppen, Villano, Doignon, and Johannesen (1990), however, the scope of this theory was extended by bringing into the picture the underlying skills and capabilities that are relevant to solving the problems (see Albert, Schrepp, and Held (1994) and Lukas and Albert (1993)). Most of the theoretical results on the extended framework are due to Doignon (1994); see also Doignon and Falmagne (1999), and Düntsch and Gediga (1995). An independent, but similar approach has been suggested by Korossy (1997, 1999). Identifying the skills that are sufficient for solving each of the considered problems provides a complete characterization of the solution behavior (ignoring response errors such as lucky guesses or careless errors for the moment). For each subset of skills an individual may be equipped with, the set of problems that can be solved within it is uniquely defined. This subset of problems constitutes a possible knowledge state, and their collection forms a knowledge structure in the sense of Doignon and Falmagne (1985, 1999).

A major challenge to this approach comes from the need to integrate distributed information on (partially) overlapping domains and skill sets, which arises in a number of scenarios. Consider a situation in which various experts independently identify and assign relevant skills to the problems in a domain. In realistic applications, where the number of problems tends to be very large, the experts may not be able to cover the whole set, but only subsets with sufficient overlap. In any case, most certainly the expert assignments will not completely coincide. The question, then, is whether the assignments are at least consistent, so that the local information provided by the experts can be aggregated into a global skill assignment. Similar problems emerge when implementing the skill-based extension of knowledge space theory in technology-enhanced learning (cf. Conlan, Hockemeyer, Wade, and Albert (2002)) within a distributed and open systems architecture. In these

* Corresponding address: Psychological Institute, University of Tübingen, Friedrichstr. 21, 72072 Tübingen, Germany.
E-mail address: juergen.heller@uni-tuebingen.de (J. Heller).

0022-2496/$ - see front matter © 2008 Published by Elsevier Inc.
doi:10.1016/j.jmp.2008.01.003

systems the services as well as the resources (e.g. repositories of problems) they access may reside at different locations. Moreover, the local problem repositories need not be static but may be modified by adding or removing problems. Again, the question is whether the skills assigned to the problems (through appropriate metadata tags) in those repositories can be integrated in a consistent way.

The present paper provides answers to these questions. After recalling some basic concepts from the theory of knowledge spaces and its skill-based extension, Section 3 introduces the notion of a distributed skill function. Its theoretical treatment is prepared by generalizing and extending previous results on the meshing of knowledge structures (Falmagne & Doignon, 1998) in Section 4. Finally, Section 5 provides the central results linking consistent aggregation of the information in a distributed skill function to the meshing operation. The paper concludes with a discussion of the impact of these results and indicates their relevance for practical applications.

2. Preliminaries

In this section we briefly introduce the basic concepts and results on which the subsequent sections are built upon (cf. Doignon and Falmagne (1999) and Düntsch and Gediga (1995)).

Throughout this paper the basic sets Q and S (also if subscripted) are assumed to be nonempty and finite, unless stated otherwise. The set Q is called a *knowledge domain* and refers to problems (exercises, questions, etc.), while the set S collects the skills that are assumed to be relevant for solving the problems in Q.

Definition 1. A *knowledge structure* is a pair $(Q, \mathcal{K})$ where $\mathcal{K}$ is a family of subsets of Q, containing at least Q and the emptyset $\emptyset$. The elements K of $\mathcal{K}$ are called *(knowledge) states*.

A knowledge state is identified with the subset of problems a person is capable of solving, and the set of possible knowledge states for the given domain constitutes a knowledge structure. Whenever no confusions can arise we will simply use $\mathcal{K}$ to refer to the knowledge structure. The knowledge structure $\mathcal{K}$ is called a *knowledge space* if it is closed for union, a *closure space* if it is closed for intersection, and a *quasi-ordinal space* if it is closed for union as well as for intersection.

Definition 2. Let $(Q, \mathcal{K})$ be a knowledge structure, let A be some nonempty subset of Q and $\mathcal{H} = \{H \in 2^A \mid H = A \cap K \text{ for some } K \in \mathcal{K}\}$. Then $(A, \mathcal{H})$ is called a *substructure* of the *parent structure* $(Q, \mathcal{K})$. Sometimes the substructure $\mathcal{H}$ is referred to as the *trace* of $\mathcal{K}$ on A, and is denoted by $\mathcal{K}|_A$.

The notion of a substructure of a knowledge structure will play a prominent role in the following. The central concept of this paper, however, is that of a skill function.

Definition 3. A *skill function* is a triple (Q, S, μ), where μ is a mapping from Q to 2^{2^S} such that each $\mu(q)$, $q \in Q$, is a nonempty collection of nonempty pairwise incomparable subsets of S (with respect to set inclusion). The elements $C \in \mu(q)$ are called *competencies*.

More than one competency may be assigned to a problem, representing the fact that there may be more than one way to solve it. The skills contained in each competency are assumed to be minimally sufficient for solving the problem. This minimality motivates the assumption that competencies are pairwise incomparable. Thus, this property will be called *minimality condition*. Again, whenever the basic sets Q and S are clear from the context, we simply use μ to refer to the respective skill function. Notice that each skill function (Q, S, μ) induces a mapping $p\colon 2^S \to 2^Q$ defined by

$$p(T) = \{q \in Q \mid \text{there is a } C \in \mu(q) \text{ such that } C \subseteq T\}$$

for all $T \subseteq S$, which is a so-called problem function.

Definition 4. A *problem function* p is a mapping from 2^S to 2^Q that is monotonic with respect to set inclusion, and satisfies $p(\emptyset) = \emptyset$ and $p(S) = Q$.

The problems in $p(T)$ are exactly those that can be solved within the subset T of skills. Thus, the range of the problem function p forms a knowledge structure $(Q, \mathcal{K})$ consisting of the knowledge states that are possible given the skill function μ. The knowledge structure $(Q, \mathcal{K})$ is said to be *delineated* by the skill function μ.

Assigning the induced problem function to the corresponding skill function defines a mapping. According to Düntsch and Gediga (1995), for a given knowledge domain Q and skill set S, this mapping actually forms a bijection between the set of all skill functions and the set of all problem functions. Both notions thus are equivalent.

3. Distributed skill functions

Let us now turn to the situation where a collection of skill functions (Q_i, S_i, μ_i) is given, with i an element of a finite or infinite index set I. Assume the most general case, in which knowledge domains Q_i as well as skill sets S_i may show (partial) overlap. Notice that the set S_i may also be taken as the union of all competencies $C \in \mu_i(q)$, $q \in Q_i$. This set of actually assigned skills may differ over components, even if the skills are drawn from a common set S.

The following definition provides a straightforward construction for integrating the different skill functions into a single skill assignment defined on the union of the domains and the skill sets, respectively.

Definition 5. Given a collection of skill functions (Q_i, S_i, μ_i), $i \in I$, their *merge* (Q, S, μ) is defined by

(i) $Q = \bigcup_{i \in I} Q_i$;
(ii) $S = \bigcup_{i \in I} S_i$;
(iii) For all q in Q, $\mu(q) = \bigcup_{i \in I} \mu_i^*(q)$, with $\mu_i^*(q) = \mu_i(q)$ if q is an element of Q_i, and $\mu_i^*(q) = \emptyset$ otherwise.

By this definition μ is a mapping from Q to 2^{2^S} such that each $\mu(q)$, $q \in Q$, is a nonempty collection of nonempty subsets of S. For all $C \in \mu(q)$ there exists some j with $q \in Q_j$ and $C \in \mu_j(q)$. In other words, there is at least one skill function (Q_j, S_j, μ_j) according to which the competency C is

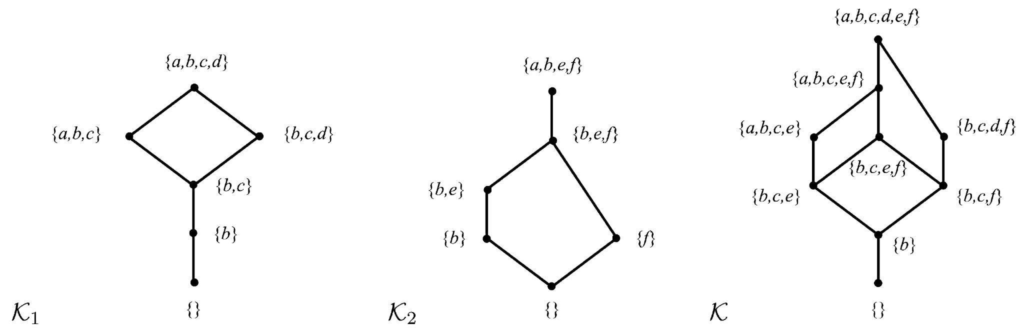

Fig. 1. Graphical representation of the knowledge structures $\mathcal{K}_1$, $\mathcal{K}_2$, and $\mathcal{K}$ of Example 1.

sufficient for solving q. Notice that the competencies in $\mu(q)$ need not satisfy the minimality condition. So, the merge of skill functions is not necessarily a skill function. Of course, from a formal point of view, the definition of a merge of skill functions may be modified so that it always results in a skill function. Violations of minimality, however, may indicate errors or inconsistencies in the definition or assignment of the competencies in the concrete application. This suggests not to make them automatically disappear, but rather to try to resolve the discrepancies in each individual case, for example, by consulting an expert.

Definition 6. Let (Q_i, S_i, μ_i), $i \in I$, be skill functions. If their *merge* (Q, S, μ) is a skill function then it is called a *distributed skill function*.

The concept of a distributed skill function was first introduced by Stefanutti, Albert, and Hockemeyer (2005). However, in their approach they confine consideration to a special case where a single competency is assigned to each problem (covered by Lemma 6), and they do not provide any theoretical results. In connection with this notion, however, a number of questions naturally arise. They are concerned with the relation between the information at the local level of the component skill functions and that at the global level captured by the distributed skill function.

i. Which conditions ensure that the global information represents a consistent aggregation of the local information?
ii. To which extent can we retrieve the local information on the components from the global information?

The problems raised by these questions have three facets. A distributed skill function μ induces a problem function $p: 2^S \to 2^Q$ in same way as the component skill functions μ_i induce the problem functions $p_i: 2^{S_i} \to 2^{Q_i}$. Moreover, μ delineates a knowledge structure $\mathcal{K}$ on the knowledge domain Q that we obtain together with the component knowledge structure $\mathcal{K}_i$ on Q_i delineated by the μ_i. Now, answering the above questions requires clarifying how the local entities μ_i, p_i and $\mathcal{K}_i$ are related to their global counterparts μ, p and $\mathcal{K}$. The following example demonstrates that these problems are far from being trivial.

Example 1. We consider two skill functions (Q_1, S_1, μ_1) and (Q_2, S_2, μ_2) which are defined by $Q_1 = \{a, b, c, d\}$, $S_1 = \{x, y, z\}$ and

$$\mu_1(a) = \{\{x, y\}, \{x, z\}\}, \qquad \mu_1(b) = \{\{x\}, \{y\}, \{z\}\},$$
$$\mu_1(c) = \{\{x\}, \{y\}\}, \qquad \mu_1(d) = \{\{y, z\}\},$$

and by $Q_2 = \{a, b, e, f\}$, $S_2 = \{w, x, y\}$ and

$$\mu_2(a) = \{\{w, y\}, \{x, y\}\}, \qquad \mu_2(b) = \{\{w\}, \{x\}\},$$
$$\mu_2(e) = \{\{x\}, \{w, y\}\}, \qquad \mu_2(f) = \{\{y\}, \{w, x\}\}.$$

They delineate the knowledge structures (see Fig. 1)

$$\mathcal{K}_1 = \{\emptyset, \{b\}, \{b, c\}, \{a, b, c\}, \{b, c, d\}, Q_1\},$$
$$\mathcal{K}_2 = \{\emptyset, \{b\}, \{f\}, \{b, e\}, \{b, e, f\}, Q_2\}.$$

Applying the above defined construction provides the distributed skill function with $Q = \{a, b, c, d, e, f\}$, $S = \{x, y, w, z\}$ and

$$\mu(a) = \{\{x, y\}, \{x, z\}, \{y, w\}\},$$
$$\mu(b) = \{\{w\}, \{x\}, \{y\}, \{z\}\}, \qquad \mu(c) = \{\{x\}, \{y\}\},$$
$$\mu(d) = \{\{y, z\}\}, \mu(e) = \{\{x\}, \{w, y\}\},$$
$$\mu(f) = \{\{y\}, \{w, x\}\}.$$

The distributed skill function μ delineates the knowledge structure

$$\mathcal{K} = \{\emptyset, \{b\}, \{b, c, e\}, \{b, c, f\}, \{a, b, c, e\}, \{b, c, d, f\}, \{b, c, e, f\}, \{a, b, c, e, f\}, Q\},$$

which is illustrated in Fig. 1, too. Now, forming the trace of $\mathcal{K}$ on Q_1 and Q_2 yields the substructures

$$\mathcal{K}|_{Q_1} = \{\emptyset, \{b\}, \{b, c, \}, \{a, b, c\}, \{b, c, d\}, Q_1\},$$
$$\mathcal{K}|_{Q_2} = \{\emptyset, \{b\}, \{b, e\}, \{b, f\}, \{a, b, e\}, \{b, e, f\}, Q_2\}.$$

This means that we have $\mathcal{K}_1 = \mathcal{K}|_{Q_1}$, but neither $\mathcal{K}_2 \subseteq \mathcal{K}|_{Q_2}$ nor $\mathcal{K}|_{Q_2} \subseteq \mathcal{K}_2$, and thus $\mathcal{K}_2 \neq \mathcal{K}|_{Q_2}$.

Example 1 shows that the local knowledge structures may not coincide with the restricted global knowledge structures.

4. Meshing knowledge structures

Falmagne and Doignon (1998) (see also Doignon and Falmagne (1999)) introduced the notion of a so-called mesh for describing the integration of two knowledge structures on the union of their domains. A knowledge structure $(Q, \mathcal{K})$ is called a *mesh* of the knowledge structures $(Q_1, \mathcal{K}_1)$ and $(Q_2, \mathcal{K}_2)$

if $Q = Q_1 \cup Q_2$, and $\mathcal{K}_1$ and $\mathcal{K}_2$ are the traces of $\mathcal{K}$ on Q_1 and Q_2, respectively. Thus, a mesh allows for recovering its component knowledge structures. It is easily seen that two knowledge structures can have one or more meshes, or no mesh at all. A necessary and sufficient condition for the existence of a mesh of two knowledge structures has been identified (see Theorem 5.14 in Doignon and Falmagne (1999). This *compatibility condition* requires that the knowledge structures $(Q_1, \mathcal{K}_1)$ and $(Q_2, \mathcal{K}_2)$ have the same trace on $Q_1 \cap Q_2$. For explicitly constructing a mesh of two compatible knowledge structures, Falmagne and Doignon (1998) provide the following definition.

Definition 7. Let $(Q_1, \mathcal{K}_1)$ and $(Q_2, \mathcal{K}_2)$ be two compatible knowledge structures. The knowledge structure $(Q_1 \cup Q_2, \mathcal{K}_1 \star \mathcal{K}_2)$ defined by

$$\mathcal{K}_1 \star \mathcal{K}_2 = \{K \in 2^{Q_1 \cup Q_2} \mid K \cap Q_1 \in \mathcal{K}_1,\ K \cap Q_2 \in \mathcal{K}_2\}$$

is the *maximal mesh* of $\mathcal{K}_1$ and $\mathcal{K}_2$.

The term maximal mesh refers to the fact that we have $\mathcal{K} \subseteq \mathcal{K}_1 \star \mathcal{K}_2$ for any mesh $\mathcal{K}$ of $\mathcal{K}_1$ and $\mathcal{K}_2$. Notice that an equivalent definition of the maximal mesh is

$$\mathcal{K}_1 \star \mathcal{K}_2 = \{K_1 \cup K_2 \mid K_1 \in \mathcal{K}_1, K_2 \in \mathcal{K}_2, \text{ and } K_1 \cap Q_2 = K_2 \cap Q_1\}.$$

The maximal meshing operator $\star$ has some interesting properties. While commutativity of $\star$ is immediate from the definition, its associativity, given the respective pairs of knowledge structures are meshable, has been shown by Falmagne and Doignon (1998). More precisely, let $(Q_1, \mathcal{K}_1), (Q_2, \mathcal{K}_2), (Q_3, \mathcal{K}_3)$ be knowledge structures such that the four pairs $(\mathcal{K}_1, \mathcal{K}_2)$, $(\mathcal{K}_1 \star \mathcal{K}_2, \mathcal{K}_3)$, $(\mathcal{K}_2, \mathcal{K}_3)$, $(\mathcal{K}_1, \mathcal{K}_2 \star \mathcal{K}_3)$ are compatible. Then we have

$$(\mathcal{K}_1 \star \mathcal{K}_2) \star \mathcal{K}_3 = \mathcal{K}_1 \star (\mathcal{K}_2 \star \mathcal{K}_3)$$

(see Doignon and Falmagne (1999, Theorem 5.24)). We generalize the notion of a mesh to a finite or even infinite number of knowledge structures. As already introduced above, let I denote a finite or infinite index set.

Definition 8. The knowledge structure $(Q, \mathcal{K})$ is called a *mesh* of the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, if

(i) $Q = \bigcup_{i \in I} Q_i$;
(ii) $\mathcal{K}_i = \mathcal{K}|_{Q_i}$ for all $i \in I$.

The construction of a maximal mesh may be generalized accordingly. Whenever there exists a mesh of the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, we obtain the *maximal mesh* by defining

$$\mathcal{K}^\star = \left\{K \in 2^{\bigcup_{i \in I} Q_i} \mid K \cap Q_i \in \mathcal{K}_i, \text{ for all } i \in I\right\}.$$

Definition 9. A collection $(Q_i, \mathcal{K}_i)$, $i \in I$, of knowledge structures is said to be *pairwise compatible* if any two of the knowledge structures have the same trace on the intersection of their domains, i.e. $\mathcal{K}_j|_{Q_j \cap Q_k} = \mathcal{K}_k|_{Q_j \cap Q_k}$ for all $j, k \in I$.

A collection of knowledge states K_i, $i \in I$, with $K_i \in \mathcal{K}_i$ is said to be *pairwise compatible* if for all $j, k \in I$ we have $K_j \cap Q_k = K_k \cap Q_j$.

Pairwise compatibility still remains to be a necessary condition for the existence of a mesh of the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, also if $|I| > 2$.

Lemma 1. *If the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, are meshable then the knowledge structures are pairwise compatible.*

Proof. Let $(Q, \mathcal{K})$ denote a mesh of the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$. Consider the knowledge structures $(Q_j, \mathcal{K}_j)$ and $(Q_k, \mathcal{K}_k)$ for arbitrary $j, k \in I$, and suppose that $K_j \in \mathcal{K}_j$. Then by definition there is a state $K \in \mathcal{K}$ such that $K \cap Q_j = K_j$. With $K_k = K \cap Q_k \in \mathcal{K}_k$ we obtain

$$K_j \cap Q_k = K \cap Q_j \cap Q_k = K \cap Q_k \cap Q_j = K_k \cap Q_j.$$

This means that $\mathcal{K}_j|_{Q_j \cap Q_k} \subseteq \mathcal{K}_k|_{Q_j \cap Q_k}$, and the converse inclusion follows by symmetry. □

In the case of $|I| > 2$, however, pairwise compatibility is no longer sufficient for the existence of a mesh, as the following example demonstrates.

Example 2. Consider the knowledge structures

$$Q_1 = \{a, b\}, \qquad \mathcal{K}_1 = \{\emptyset, \{a\}, \{a, b\}\};$$
$$Q_2 = \{b, c\}, \qquad \mathcal{K}_2 = \{\emptyset, \{b\}, \{b, c\}\};$$
$$Q_3 = \{a, c\}, \qquad \mathcal{K}_3 = \{\emptyset, \{c\}, \{a, c\}\}.$$

It is easily seen that $\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3$ are pairwise compatible as all pairs have equal traces on the intersection of their respective domains. Thus, we can construct the maximal mesh

$$\mathcal{K}_1 \star \mathcal{K}_2 = \{\emptyset, \{a\}, \{a, b\}, \{a, b, c\}\}$$

on $Q_1 \cup Q_2 = \{a, b, c\}$, which is the unique mesh of $\mathcal{K}_1$ and $\mathcal{K}_2$. The trace

$$(\mathcal{K}_1 \star \mathcal{K}_2)|_{\{a,c\}} = \{\emptyset, \{a\}, \{a, c\}\}$$

on $(Q_1 \cup Q_2) \cap Q_3 = \{a, c\}$, however, is *not* equal to $\mathcal{K}_3|_{\{a,c\}} = \mathcal{K}_3$. Thus, there is no mesh of the knowledge structures $\mathcal{K}_1 \star \mathcal{K}_2$ and $\mathcal{K}_3$ on $Q_1 \cup Q_2 \cup Q_3 = \{a, b, c\}$.

Lemma 2. *Let the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, be meshable. Consider the knowledge structure*

$$\mathcal{K}' = \left\{\bigcup_{i \in I} K_i \mid K_i \in \mathcal{K}_i \text{ for all } i \in I, \text{ and } K_j \cap Q_k = K_k \cap Q_j \text{ for all } j, k \in I\right\}.$$

Then $\mathcal{K}'$ equals the maximal mesh $\mathcal{K}^\star$ of the knowledge structures $\mathcal{K}_i$, $i \in I$.

Proof. Let $K \in \mathcal{K}^\star$, and define $K_i = K \cap Q_i$ for all $i \in I$. Then $K = \cup_{i \in I} K_i$ and for all $j, k \in I$, we have $K_j \cap Q_k =$

$K \cap Q_j \cap Q_k = K_k \cap Q_j$. Thus, we conclude $K \in \mathcal{K}'$. Conversely, let $K \in \mathcal{K}'$, then for each $i \in I$

$$K \cap Q_i = \left(\bigcup_{j \in I} K_j\right) \cap Q_i = \bigcup_{j \in I}(K_j \cap Q_i)$$
$$= \bigcup_{j \in I}(K_i \cap Q_j) = K_i \cap \left(\bigcup_{j \in I} Q_j\right)$$
$$= K_i \in \mathcal{K}_i,$$

and thus $K \in \mathcal{K}^\star$. □

Notice that in the case of a finite index set it is easily seen that the maximal mesh $\mathcal{K}^\star$ coincides with the result of a repeated application of the binary maximal meshing operator, i.e. for $I = \{1, \dots, n\}$ we have

$$\mathcal{K}^\star = \mathcal{K}_1 \star \cdots \star \mathcal{K}_n.$$

Now, returning to the general situation where I may be infinite, Proposition 1 provides a necessary and sufficient condition for the meshability of any arbitrary collection of knowledge structures.

Proposition 1. *For any collection of knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, the following statements are equivalent.*

(i) *The knowledge structures are meshable.*
(ii) *For all $j \in I$ and all $K_j \in \mathcal{K}_j$ there exists a collection K_i', $i \in I$, of pairwise compatible knowledge states $K_i' \in \mathcal{K}_i$ such that $K_j' = K_j$.*

Proof. First we show that (i) implies (ii). Consider the maximal mesh $\mathcal{K}^\star$ of $(Q_i, \mathcal{K}_i)$, $i \in I$, which exists by assumption. Let $K_j \in \mathcal{K}_j$. Then there is $K \in \mathcal{K}^\star$ such that $K \cap Q_j = K_j$. Thus (ii) holds with $K_i' = K \cap Q_i$ for all $i \in I$ by Lemma 2. Conversely, assume the conditions stated in (ii). We define the knowledge structure

$$\mathcal{K} = \{K \in 2^{\bigcup_{i \in I} Q_i} \mid K \cap Q_i \in \mathcal{K}_i, \text{ for all } i \in I\}$$

on $\cup_{i \in I} Q_i$ and show that it is a mesh. Clearly, we have $\mathcal{K}|_{Q_i} \subseteq \mathcal{K}_i$ for all $i \in I$. For proving the reverse inclusion let $j \in I$ be arbitrary and consider any state $K_j \in \mathcal{K}_j$. Then, according to (ii) there is a collection of pairwise compatible knowledge states $K_i' \in \mathcal{K}_i$, $i \in I$ with $K_j' = K_j$. Taking the union $K = \cup_{i \in I} K_i'$ we obtain for all $k \in I$

$$K \cap Q_k = \bigcup_{i \in I}(K_i' \cap Q_k) = \bigcup_{i \in I}(K_k' \cap Q_i)$$
$$= K_k' \cap \left(\bigcup_{i \in I} Q_i\right) = K_k'.$$

This means that $K \in \mathcal{K}$. Moreover, for $k = j$ we obtain $K_j = K_j' = K \cap Q_j \in \mathcal{K}|_{Q_j}$ and, as $j \in I$ and $K_j \in \mathcal{K}_j$ were arbitrary, this provides $\mathcal{K}_i \subseteq \mathcal{K}|_{Q_i}$ for all $i \in I$. □

Proposition 1 shows that meshability of a collection of knowledge structures is equivalent to requiring that each of their states can be "covered" by a collection of pairwise compatible knowledge states in the sense of Definition 9.

In the following, we present some additional sets of conditions for the meshability of knowledge structures.

Lemma 3. *For any collection of knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$ the following statements are equivalent.*

(i) *For all $j, k \in I$ we have $\mathcal{K}_j|_{Q_j \cap Q_k} \subseteq \mathcal{K}_k$;*
(ii) *The knowledge structures are pairwise compatible and for all $j, k \in I$ we have $\mathcal{K}_j|_{Q_j \cap Q_k} \subseteq \mathcal{K}_j$.*

Proof. The equivalence of (i) to the property $\mathcal{K}_j|_{Q_j \cap Q_k} \subseteq \mathcal{K}_j$ for all $j, k \in I$ given $\mathcal{K}_j$ and $\mathcal{K}_k$ are compatible is obvious because compatibility means that we have $\mathcal{K}_j|_{Q_j \cap Q_k} = \mathcal{K}_k|_{Q_j \cap Q_k}$. Thus, it remains to show that pairwise compatibility follows from (i). The latter property, however, implies that for any $K_j \in \mathcal{K}_j$ there is a $K_k \in \mathcal{K}_k$ such that $K_j \cap (Q_j \cap Q_k) = K_k$. Taking the intersection with Q_j on both sides of this equation then yields $K_j \cap Q_k = K_k \cap Q_j$, which, by symmetry, provides compatibility. □

With Lemma 3 we obtain the following corollary to Proposition 1.

Corollary 1. *Let $(Q_i, \mathcal{K}_i)$, $i \in I$ be knowledge structures that satisfy the equivalent conditions of Lemma 3. Then the knowledge structures are meshable.*

Proof. Let $j \in I$ and K_j be a state of $\mathcal{K}_j$. Condition (i) of Lemma 3 implies that for all $i \in I$ we have $K_j \cap Q_i = K_i'$ for some $K_i' \in \mathcal{K}_i$. The collection K_i', $i \in I$, of these knowledge states satisfies the conditions in (ii) of Proposition 1. We have the identity $K_j' = K_j$ and for all $k, l \in I$ the equation $K_k' \cap Q_l = (K_j \cap Q_k) \cap Q_l = (K_j \cap Q_l) \cap Q_k = K_l' \cap Q_k$, which provides pairwise compatibility of the K_i'. □

Moreover, the equivalent conditions of Lemma 3 admit an even stricter conclusion if we confine consideration to knowledge spaces. Doignon and Falmagne (1999, Theorem 5.20) show that in the case of two knowledge spaces these conditions warrant that the mesh is inclusive (see Definition 10). This result can be generalized to the present situation.

Definition 10. The mesh $(Q, \mathcal{K})$ of the knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, is called *inclusive*, if $\bigcup_{i \in I} K_i \in \mathcal{K}$ for all $K_i \in \mathcal{K}_i$, $i \in I$.

Proposition 2. *Let $(Q_i, \mathcal{K}_i)$, $i \in I$, be knowledge spaces that satisfy the equivalent conditions of Lemma 3. Then these knowledge spaces admit some inclusive mesh.*

Proof. There exists a maximal mesh $(\cup_{i \in I} Q_i, \mathcal{K}^\star)$ by Corollary 1. Setting $K = \bigcup_{i \in I} K_i$ for some arbitrary $K_i \in \mathcal{K}_i$, and choosing any index $j \in I$ we obtain

$$K \cap Q_j = \bigcup_{i \in I}(K_i \cap Q_j).$$

Noticing that $K_i \cap Q_j = K_i \cap (Q_i \cap Q_j)$, condition (i) of Lemma 3 then implies that these elements all are in $\mathcal{K}_j$. Since $\mathcal{K}_j$ is assumed to be a knowledge space (i.e. closed for union) this finally provides that $K \cap Q_j \in \mathcal{K}_j$. Consequently, K is a state in $\mathcal{K}^\star$, which shows that $\mathcal{K}^\star$ is inclusive. □

The following definition introduces a kind of regularity condition for the compatibility of knowledge structures.

Definition 11. The knowledge structures $(Q_i, \mathcal{K}_i), i \in I$ are called *strictly compatible* if they are pairwise compatible and for all indices $j, k, l \in I$ and all $K_j \in \mathcal{K}_j$, $K_k \in \mathcal{K}_k$, $K_l \in \mathcal{K}_l$

$$K_j \cap Q_k = K_k \cap Q_j \quad \text{and} \quad K_k \cap Q_l = K_l \cap Q_k$$

imply

$$K_j \cap Q_l = K_l \cap Q_j.$$

In order to motivate this notion define a binary relation $R_{jk} \subseteq \mathcal{K}_j \times \mathcal{K}_k$ by

$$K_j \, R_{jk} \, K_k \quad \text{iff} \quad K_j \cap Q_k = K_k \cap Q_j$$

for all $j, k \in I$ and $K_j \in \mathcal{K}_j$, $K_k \in \mathcal{K}_k$. Thus, the states K_j and K_k are in relation R_{jk} if they both induce the same subset when restricted to the intersection $Q_j \cap Q_k$. Strict compatibility of the knowledge structures $(Q_i, \mathcal{K}_i), i \in I$, then means that for all $j, k, l \in I$ we have

$$R_{jk} R_{kl} \subseteq R_{jl},$$

where $R_{jk} R_{kl}$ denotes the (relative) product of the relations R_{jk} and R_{kl}. This means that strict compatibility can be formalized as a transitivity-like cancellation condition. The next two results identify the consequences of this structural assumption.

Proposition 3. *Any collection $(Q_i, \mathcal{K}_i)$, $i \in I$, of strictly compatible knowledge structures is meshable.*

Proof. For an arbitrary index $j \in I$ let K_j be a state of $\mathcal{K}_j$. Due to pairwise compatibility, for each $i \in I$ there is a state $K'_i \in \mathcal{K}_i$ such that $K'_i \cap Q_j = K_j \cap Q_i$. For any two of these states, say K'_k and K'_l with $k, l \in I$, strict compatibility then implies that $K'_k \cap Q_l = K'_l \cap Q_k$. The knowledge states K'_i, $i \in I$ thus are pairwise compatible and satisfy $K'_j = K_j$, which in Proposition 1 was shown to be necessary and sufficient for meshability. □

The following example demonstrates that strict compatibility does not necessarily hold in every collection of meshable knowledge structures.

Example 3. Consider the knowledge structures

$$Q_1 = \{a, b\}, \qquad \mathcal{K}_1 = \{\emptyset, \{a\}, \{a, b\}\};$$
$$Q_2 = \{b, c\}, \qquad \mathcal{K}_2 = \{\emptyset, \{b\}, \{b, c\}\};$$
$$Q_3 = \{a, c\}, \qquad \mathcal{K}_3 = \{\emptyset, \{a\}, \{a, c\}\}.$$

The unique mesh of the knowledge structures $\mathcal{K}_1, \mathcal{K}_2, \mathcal{K}_3$ on $Q = Q_1 \cup Q_2 \cup Q_3 = \{a, b, c\}$ is the maximal mesh

$$\mathcal{K} = \{\emptyset, \{a\}, \{a, b\}, \{a, b, c\}\}.$$

Then the knowledge states $K_1 = \emptyset \in \mathcal{K}_1$, $K_2 = \emptyset \in \mathcal{K}_2$, and $K_3 = \{a\} \in \mathcal{K}_3$ violate strict compatibility, as we have $K_1 \cap Q_2 = K_2 \cap Q_1 = \emptyset$, $K_2 \cap Q_3 = K_3 \cap Q_2 = \emptyset$, and $\emptyset = K_1 \cap Q_3 \neq K_3 \cap Q_1 = \{a\}$.

Strict compatibility, however, follows whenever there exists some inclusive mesh.

Proposition 4. *Any collection of knowledge structures $(Q_i, \mathcal{K}_i)$, $i \in I$, that admits some inclusive mesh is strictly compatible.*

Proof. Let $(Q_i, \mathcal{K}_i), i \in I$ be knowledge structures that admit some inclusive mesh, and let $(Q, \mathcal{K}^\star)$ with $Q = \cup_{i \in I} Q_i$ be their maximal mesh. Then $\mathcal{K}^\star$ is inclusive, as it contains an inclusive mesh. Thus, for all $K_j \in \mathcal{K}_j$, $K_k \in \mathcal{K}_k$, $K_l \in \mathcal{K}_l$ with $j, k, l \in I$ the subset $K_j \cup K_k \cup K_l$ is a state of the inclusive mesh $\mathcal{K}^\star$, and strict compatibility follows from Lemma 2. □

5. Skill and problem functions

The general assumption in this section is that we are given a collection (Q_i, S_i, μ_i), $i \in I$, of skill functions, which give rise to a distributed skill function (Q, S, μ). The problem functions that correspond to the μ_i and μ are denoted by p_i and p, respectively, while the delineated knowledge structures are denoted by $\mathcal{K}_i$ and $\mathcal{K}$, respectively. For notational convenience reference to the basic sets Q_i, S_i as well as Q, S is omitted whenever possible. The following Lemma is immediate from Definitions 4 and 5.

Lemma 4. *For all $i \in I$ the following inclusions hold.*

(i) $\mu_i(q) \subseteq \mu(q) \cap 2^{S_i} \subseteq \mu(q)$ *for all* $q \in Q_i$;
(ii) $p_i(T \cap S_i) \subseteq p(T \cap S_i) \cap Q_i \subseteq p(T) \cap Q_i$ *for all subsets* $T \subseteq S$.

Moreover, we have

(iii) $p(T) = \cup_{i \in I} p_i(T \cap S_i)$ *for all subsets* $T \subseteq S$.

Lemma 4(iii) shows that the problem function p induced by the distributed skill function μ can be computed directly from the component problem functions p_i by taking their union. This has consequences for the properties that p inherits from the p_i, as we will see below. The sequences of inclusions in Lemma 4(i) and (ii) give rise to interesting special cases, in which one or all of the corresponding inclusions are restricted to equality.

Proposition 5. *For all $i \in I$ the following statements are equivalent.*

(i) $\mu_i(q) = \mu(q) \cap 2^{S_i}$ *for all* $q \in Q_i$;
(ii) $p_i(T \cap S_i) = p(T \cap S_i) \cap Q_i$ *for all subsets* $T \subseteq S$.

Moreover, both statements imply $\mathcal{K}_i \subseteq \mathcal{K}|_{Q_i}$.

Proof. Due to Lemma 4 we only have to show that $p(T \cap S_i) \cap Q_i \subseteq p_i(T \cap S_i)$ for all subsets $T \subseteq S$ in order to prove the implication from (i) to (ii). So, let $T \subseteq S$ and consider $q \in p(T \cap S_i) \cap Q_i$. Then there is a $C \subseteq T \cap S_i$ such that $C \in \mu(q) \cap 2^{S_i} = \mu_i(q)$. This, however, means that $q \in p_i(T \cap S_i)$.

According to Lemma 4 we only have to prove that $\mu(q) \cap 2^{S_i} \subseteq \mu_i(q)$ for showing that (ii) implies (i). So, let $C \subseteq S_i$ and $q \in Q_i$. Then $C \in \mu(q) \cap 2^{S_i}$ implies that $q \in p(C) \cap Q_i = p_i(C)$. Thus, there is a $C' \subseteq C$ such that $C' \in \mu_i(q)$. With

$\mu_i(q) \subseteq \mu(q) \cap 2^{S_i}$ by Lemma 4 the assertion then follows because $C = C'$ by minimality.

In order to show that (ii) implies $\mathcal{K}_i \subseteq \mathcal{K}|_{Q_i}$, let K_i be an arbitrary state in $\mathcal{K}_i$. Then there is a subset of skills $T \subseteq S$ such that $K_i = p_i(T \cap S_i)$. By assumption we get $K_i = p_i(T \cap S_i) = p(T \cap S_i) \cap Q_i$. Since $p(T \cap S_i)$ is in $\mathcal{K}$, it follows that $K_i \in \mathcal{K}|_{Q_i}$. □

In general, reversing the implication in Proposition 5 is not possible, as the following counter-example demonstrates.

Example 4. Let the skill functions (Q_1, S_1, μ_1) and (Q_2, S_2, μ_2) be defined as in Example 1 except for $\mu_2(f) = \{\{w, x\}\}$. Then it is easily checked that $\mathcal{K}_2 \subseteq \mathcal{K}|_{Q_2}$ (in fact we have $\mathcal{K}_2 = \mathcal{K}|_{Q_2}$), but there is $\emptyset = p_2(\{y\}) \neq p(\{y\}) \cap Q_2 = \{b\}$.

Now, Proposition 6 considers the case where for each problem $q \in Q_i$ all the competencies in $\mu(q)$ are actually subsets of S_i.

Proposition 6. *For all $i \in I$ the following statements are equivalent.*

(i) $\mu(q) \cap 2^{S_i} = \mu(q)$ *for all* $q \in Q_i$;
(ii) $p(T \cap S_i) \cap Q_i = p(T) \cap Q_i$ *for all subsets* $T \subseteq S$.

Proof. For proving the implication from (i) to (ii) we assume that for every problem $q \in Q_i$, each competency $C \in \mu(q)$ is a subset of S_i. According to Lemma 4 it remains to show that $p(T) \cap Q_i \subseteq p(T \cap S_i) \cap Q_i$. Let T be an arbitrary subset of S and $q \in p(T) \cap Q_i$. In other words, there is a competency $C \in \mu(q)$ with $C \subseteq T$. The assumption implies that $C \subseteq T \cap S_i$. Consequently, we have $q \in p(T \cap S_i) \cap Q_i$.

The converse implication is proved by contradiction. Assume that there is a problem q in Q_i which can be solved by a competency $C \in \mu(q)$ with $C \nsubseteq S_i$. Then $q \in p(C) \cap Q_i$, and by assumption we get $q \in p(C \cap S_i) \cap Q_i$. Hence, there must be a competency $C' \in \mu(q)$ with $C' \subseteq C \cap S_i \subset C$, which contradicts the minimality condition. □

We may now combine the above results to obtain the following proposition.

Proposition 7. *For all $i \in I$ the following statements are equivalent.*

(i) $\mu_i(q) = \mu(q)$ *for all* $q \in Q_i$;
(ii) $p_i(T \cap S_i) = p(T) \cap Q_i$ *for all subsets* $T \subseteq S$.

Moreover, both statements imply $\mathcal{K}_i = \mathcal{K}|_{Q_i}$.

Proof. The equivalence of (i) and (ii) follows from Propositions 5 and 6. To see that $\mathcal{K}_i = \mathcal{K}|_{Q_i}$ is implied by (ii) we only have to observe that for each $i \in I$

$$\mathcal{K}_i = \{p_i(T \cap S_i) \mid T \subseteq S\} = \{p(T) \cap Q_i \mid T \subseteq S\} = \mathcal{K}|_{Q_i}.$$
□

Condition (i) of Proposition 7 formalizes the most stringent notion of consistency that a collection of skill functions may satisfy: Whenever the domains of any skill functions overlap then they all assign the same competencies to the common problems. Proposition 7 asserts that in this case the knowledge structure $\mathcal{K}$ delineated by the distributed skill function μ is a mesh of the knowledge structures $\mathcal{K}_i$ delineated by the component skill functions μ_i. The converse of this statement is not true, as is shown by Example 4 above. This gives room for the following definition characterizing consistency of skill functions through meshability of their induced knowledge structures.

Definition 12. A collection of skill functions (Q_i, S_i, μ_i), $i \in I$, is called *consistent* whenever the collection of their delineated knowledge structures $\mathcal{K}_i$, $i \in I$, is meshable.

As the knowledge structure delineated by a skill function is nothing but the range of its associated problem function, we can arrive at a characterization of consistent collections of skill functions by the subsequent corollary to Proposition 1. We prepare this result by introducing the following notion.

Definition 13. Let (Q_i, S_i, μ_i), $i \in I$, be a collection of skill functions. A collection of skill sets T_i, $i \in I$, with $T_i \subseteq S_i$ is called *pairwise compatible* if the following equivalent conditions hold.

(i) For all $j, k \in I$ we have $p_j(T_j) \cap Q_k = p_k(T_k) \cap Q_j$;
(ii) For all $j, k \in I$ and all $q \in Q_j \cap Q_k$ there is a $C_j \in \mu_j(q)$ with $C_j \subseteq T_j$ if and only if there is a $C_k \in \mu_k(q)$ with $C_k \subseteq T_k$.

Definition 13 recasts the notion of a collection of pairwise compatible knowledge states (Definition 9) in terms of skill sets. Notice that the second condition results from simply plugging in the definition of the problem function into the first one.

Corollary 2. *A collection of skill functions (Q_i, S_i, μ_i), $i \in I$, is consistent if and only if for all $j \in I$ and any skill set $T_j \subseteq S_j$ there exists a collection T_i', $i \in I$, of pairwise compatible skill sets $T_i' \subseteq S_i$ such that $p_j(T_j') = p_j(T_j)$.*

Corollary 2 shows that consistency of a collection of skill functions is equivalent to requiring that each "local" skill set can be "covered" by a collection of pairwise compatible skill sets in the sense of Definition 13.

We conclude this section by treating several special cases. The following corollary to Propositions 5 and 7 considers the case of pairwise disjoint problem domains and skill sets.

Corollary 3. *Consider a collection of skill functions (Q_i, S_i, μ_i), $i \in I$, and the corresponding distributed skill function (Q, S, μ).*

(i) *If the problem domains Q_i are pairwise disjoint then the knowledge structure $\mathcal{K}$ delineated by μ is a mesh of the component knowledge structures $\mathcal{K}_i$, i.e. $\mathcal{K}_i = \mathcal{K}|_{Q_i}$ for all $i \in I$.*
(ii) *If the skill domains S_i are pairwise disjoint then $\mathcal{K}_i \subseteq \mathcal{K}|_{Q_i}$ for all $i \in I$.*

Knowledge spaces, closure spaces, and quasi-ordinal spaces were introduced as knowledge structures that are closed for union, for intersection, and for both union and intersection, respectively. In general, a knowledge structure delineated by a skill function does not necessarily satisfy any of these properties. It can be shown that each arbitrary knowledge structure is delineated by at least one skill function (cf. Theorem 3.1 of Düntsch and Gediga (1995) and Theorem 4.18 of Doignon and Falmagne (1999)). Subsequently we consider special cases of skill and problem functions that delineate knowledge spaces and closure spaces, respectively, and that have already received some attention. The following two lemmas parallel a result of Gediga and Düntsch (2002, Theorem 3.1), but rest on somewhat more direct proofs.

Lemma 5 links the disjunctive model of Doignon and Falmagne (1999), which is equivalent to characterizing the skill function as in condition (i), to the case where the problem function preserves union as defined in condition (ii), which is treated by Düntsch and Gediga (1995). In general, of course, a problem function p will not preserve union. The reason for this is that the combination of two subsets of skills may enable an individual to solve additional problems which cannot be solved given either of the subsets only. This fact is captured by the inclusion $p(T_1) \cup p(T_2) \subseteq p(T_1 \cup T_2)$ holding for all $T_1, T_2 \in S$. Notice that the conditions in Lemma 5 are not necessary for the delineated knowledge structure being closed for union (see Düntsch and Gediga (1995), for a counter-example).

Lemma 5. *Let $\mu: Q \to 2^{2^S}$ be a skill function, p its induced problem function, and $\mathcal{K}$ its delineated knowledge structure. Then the following statements are equivalent.*

(i) *For all $q \in Q$ each of the competencies $C \in \mu(q)$ is a singleton set;*
(ii) $p(T_1 \cup T_2) = p(T_1) \cup p(T_2)$ *for all $T_1, T_2 \in S$.*

Moreover, these conditions imply that $\mathcal{K}$ is a knowledge space.

Proof. Since the monotonicity of p provides $p(T_1) \cup p(T_2) \subseteq p(T_1 \cup T_2)$, the crucial inclusion in (ii) is $p(T_1 \cup T_2) \subseteq p(T_1) \cup p(T_2)$ for all $T_1, T_2 \in S$. So, assume (i) and $q \in p(T_1 \cup T_2)$. Then there is a singleton set $C \in \mu(q)$ such that $C \subseteq T_1 \cup T_2$. This means that $C \subseteq T_1$ or $C \subseteq T_2$ providing $q \in p(T_1) \cup p(T_2)$. Conversely, assume (ii) and $|C| > 1$ for some $C \in \mu(q)$ and $q \in Q$. Then there is a partition $C = T_1 \cup T_2$ with nonempty and disjoint subsets $T_1, T_2 \subseteq S$. Thus, we infer $q \in p(T_1 \cup T_2)$, and by (ii) obtain that $q \in p(T_1)$ or $q \in p(T_2)$. In the first case there is a $C' \in \mu(q)$ such that $C' \subseteq T_1 \subset C$, which contradicts the minimality condition. The second case is analogous.

Applying Theorem 4.4 of Doignon and Falmagne (1999) drawing upon condition (i) or, equivalently, Proposition 2.4 of Düntsch and Gediga (1995) drawing upon condition (ii) yields that $\mathcal{K}$ is a knowledge space. □

The skill functions satisfying condition (i) of the following Lemma 6 define a conjunctive model in the sense of Doignon and Falmagne (1999). In this case the problem function p not only satisfies the inclusion $p(T_1 \cap T_2) \subseteq p(T_1) \cap p(T_2)$ for all $T_1, T_2 \in S$, but preserves intersections.

Lemma 6. *Let $\mu: Q \to S$ be a skill function, p its induced problem function, and $\mathcal{K}$ its delineated knowledge structure. Then the following statements are equivalent.*

(i) *For all $q \in Q$ we have $\mu(q) = \{C\}$ for some $C \subseteq S$;*
(ii) $p(T_1 \cap T_2) = p(T_1) \cap p(T_2)$ *for all $T_1, T_2 \in S$.*

Moreover, these conditions imply that $\mathcal{K}$ is a closure space.

Proof. The critical inclusion in (ii) is $p(T_1) \cap p(T_2) \subseteq p(T_1 \cap T_2)$ for all $T_1, T_2 \in S$, as the converse inclusion follows from the monotonicity of p. So, assume (i) and $q \in p(T_1) \cap p(T_2)$. As there is only a single competency $C \in \mu(q)$ we have $C \subseteq T_1$ and $C \subseteq T_2$. This means that $C \subseteq T_1 \cap T_2$ providing $q \in p(T_1 \cap T_2)$. Conversely, assume (ii) and that there are distinct $C_1, C_2 \in \mu(q)$ for some $q \in Q$. Then $q \in p(C_1) \cap p(C_2) = p(C_1 \cap C_2)$, and there is a $C \in \mu(q)$ such that $C \subseteq C_1 \cap C_2$, which contradicts the minimality condition.

Theorem 4.14 of Doignon and Falmagne (1999) drawing upon condition (i) provides that $\mathcal{K}$ is a closure space. □

As in the case of closure for union treated in Lemma 5, the conditions stated in Lemma 6 are not necessary for the delineated knowledge structure to be closed for intersection. Take $Q = \{a, b\}$ and $S = \{x, y, z, w\}$ with $\mu(a) = \{\{x, y\}\}$ and $\mu(b) = \{\{x, z\}, \{w\}\}$ as a counter-example.

Even if each of the component skill functions μ_i delineates a knowledge space (closure space, quasi-ordinal space, respectively) the distributed skill function μ does not necessarily delineate a knowledge space (closure space, quasi-ordinal space, respectively) as well. It is easy to provide counter-examples. Thus, in general, properties of the component structures $\mathcal{K}_i$ are not passed on to $\mathcal{K}$. However, it is obvious from its definition that a distributed skill function μ will satisfy condition (i) of Lemma 5 whenever it holds for all the component skill functions μ_i. This also means that whenever all the p_i are homomorphisms with respect to union, so is the problem function p, and we have the following conclusion.

Corollary 4. *If the equivalent conditions of Lemma 5 hold for all components $i \in I$ then the $\mathcal{K}_i$ as well as the knowledge structure $\mathcal{K}$ delineated by the distributed skill function μ are knowledge spaces.*

Generally, if each component skill function μ_i satisfies condition (i) of Lemma 6 then this property need not hold for the distributed skill function μ, as the subsequent example demonstrates.

Example 5. We consider two skill functions (Q_1, S_1, μ_1) and (Q_2, S_2, μ_2) which are defined by $Q_1 = \{a, b, c\}$, $S_1 = \{x, y, z\}$ and

$$\mu_1(a) = \{\{x\}\}, \qquad \mu_1(b) = \{\{y, z\}\}, \qquad \mu_1(c) = \{\{z\}\},$$

and by $Q_2 = \{a, b, d, e\}$, $S_2 = \{x, y, w\}$ and

$$\mu_2(a) = \{\{w\}\}, \qquad \mu_2(b) = \{\{x\}\}, \qquad \mu_2(d) = \{\{x, y\}\},$$
$$\mu_2(e) = \{\{x\}\}.$$

It is easily checked that both sets $\{b, c\}$ and $\{a, b, e\}$ are states of the distributed knowledge structure $\mathcal{K}$, however, $\{b, c\} \cap \{a, b, e\} = \{b\}$ is not a state of $\mathcal{K}$.

The following result shows that, if the knowledge structure delineated by a distributed skill function is inclusive in the sense of Definition 10, then it is closed for union whenever the component structures $\mathcal{K}_i$ are. Notice that Proposition 8 does *not* require $\mathcal{K}$ to be a mesh of the component knowledge structures.

Proposition 8. *Let the knowledge structures $\mathcal{K}_i$ delineated by skill functions μ_i, $i \in I$, be closed for union. Furthermore, assume that for all $K_i \in \mathcal{K}_i$, $i \in I$, the union $\bigcup_{i \in I} K_i$ is a state of the knowledge structure $\mathcal{K}$ delineated by the distributed skill function μ. Then $\mathcal{K}$ is closed under union.*

Proof. Let $\mathcal{F}$ be a collection of states of $\mathcal{K}$. For each $K \in \mathcal{F}$ there is a subset of skills $T_K \subseteq S$ such that $K = p(T_K)$. Then we have the equations

$$\begin{aligned}\bigcup_{K \in \mathcal{F}} K &= \bigcup_{K \in \mathcal{F}} p(T_K) \\ &= \bigcup_{K \in \mathcal{F}} \left(\bigcup_{i \in I} p_i(T_K \cap S_i) \right) \\ &= \bigcup_{i \in I} \left(\bigcup_{K \in \mathcal{F}} p_i(T_K \cap S_i) \right),\end{aligned}$$

where the second equation rests on Lemma 4(iii).

Since the knowledge structures $\mathcal{K}_i$ are closed for union the $K_i = \bigcup_{K \in \mathcal{F}} p_i(T_K \cap S_i)$ are states in the respective $\mathcal{K}_i$. Furthermore, the union $\bigcup_{i \in I} K_i$ of the states K_i is in $\mathcal{K}$ by assumption. This means that $\bigcup_{K \in \mathcal{F}} K$ is in $\mathcal{K}$, and so $\mathcal{K}$ is closed for union. □

The converse direction is more straightforward. If the knowledge structure $\mathcal{K}$ delineated by a distributed skill function is a mesh of the component knowledge structures $\mathcal{K}_i$ then the K_i are knowledge spaces (closure spaces, quasi-ordinal spaces, respectively) whenever $\mathcal{K}$ is.

Proposition 9. *Assume that the knowledge structure $\mathcal{K}$ delineated by the distributed skill function μ is a mesh of the component knowledge structures $\mathcal{K}_i$, $i \in I$. If $\mathcal{K}$ is closed for union (closed for intersection, respectively) then each component knowledge structure $\mathcal{K}_i$ is closed for union (closed for intersection, respectively).*

Proof. The statement that each component structure is closed for union follows from Theorem 1.16 of Doignon and Falmagne (1999). We assume that $\mathcal{K}$ is closed for intersection. Let $\mathcal{F}_i$ be a family of states of the structure $\mathcal{K}_i$. Since $\mathcal{K}_i$ is equal to the structure $\mathcal{K}|_{Q_i}$ for each state $A \in \mathcal{F}_i$ there is a state B_A in $\mathcal{K}$ with $A = B_A \cap Q_i$. Now, $\bigcap_{A \in \mathcal{F}_i} A = \bigcap_{A \in \mathcal{F}_i} (B_A \cap Q_i) = (\bigcap_{A \in \mathcal{F}_i} B_A) \cap Q_i$. As $\mathcal{K}$ is a mesh of the component structures $\mathcal{K}_i$ and $\bigcap_{A \in \mathcal{F}_i} B_A$ is a state of $\mathcal{K}$, one infers that $\bigcap_{A \in \mathcal{F}_i} A$ is a state of $\mathcal{K}_i$. □

6. Conclusions

The present paper introduces the notion of a distributed skill function for treating the consistent integration of a collection of skill functions on (partially) overlapping knowledge domains. These skill functions may be conceived as representing assignments of skills to problems that come from different sources (e.g. different domain experts). It is shown that their consistency is captured by the meshability of the delineated knowledge structures. This result draws upon a characterization of the meshing of arbitrary collections of knowledge structures, which is developed in Section 4 and extends and generalizes previous results on the binary case (Doignon & Falmagne, 1999; Falmagne & Doignon, 1998). Proposition 1 provides necessary and sufficient conditions for meshability in this situation, which do not hold in general for knowledge domains with nonempty overlap. Consistency in distributed skill functions is linked to the meshability of the delineated knowledge structures by Proposition 7. Corollary 2 then identifies properties of a collection of skill functions that are necessary and sufficient for their consistency.

We now discuss the impact of the results on the two scenarios outlined in the introduction, which are the integration of expert skill assignments and technology-enhanced learning within a distributed systems architecture. Let us first consider knowledge assessment in the latter case, where resources residing at different locations may be contributed by possibly different authors. The skill assignments in these local problem repositories are implemented through appropriate metadata tags (e.g. Conlan et al. (2002)), and the distributed skill function represents their aggregation. An individual's knowledge then may be assessed at two different levels: At a local level based on one of the local problem repositories and its associated knowledge structure, and at a global level through the knowledge structure delineated by the distributed skill function. The following questions arise in this context.

(i) Can an assessment at the global level build upon the information collected within a local assessment?
(ii) Is it possible to localize a given global assessment to any of the local domains?

The subsequent example indicates that within the considered setting this kind of smooth transition between local and global knowledge assessment is mediated by the consistency introduced above.

In order to assess their knowledge individuals are presented with a sequence of problems. For each of these problems q it is recorded whether it is solved or not and let these two cases be denoted by q and $\overline{q}$, respectively. We also use the notation $\mathcal{K}_q = \{K \in \mathcal{K} \mid q \in K\}$ and $\mathcal{K}_{\overline{q}} = \{K \in \mathcal{K} \mid q \notin K\}$ for a knowledge structure $(Q, \mathcal{K})$. Referring to Example 1 an assessment may proceed at the local level (i.e. based on the knowledge structures $\mathcal{K}_1$ or $\mathcal{K}_2$) or at the global level (i.e. with respect to $\mathcal{K}$). Assume that assessing an individual on Q_1 provides the sequence $b\overline{d}a$ of responses. There are exactly two knowledge states in $\mathcal{K}$ that are consistent with this information. The collection of states satisfying the constraints in $b\overline{d}a$ is $\mathcal{K}_b \cap \mathcal{K}_{\overline{d}} \cap \mathcal{K}_a = \{\{a, b, c, e\}, \{a, b, c, e, f\}\}$. The state of the individual in the knowledge structure $\mathcal{K}$ on Q may be identified by continuing the assessment, i.e. by presenting problems from $Q \setminus Q_1$. Assessing the performance on problem $f \in Q_2$, for example, yields either $b\overline{d}af$ corresponding to state $\{a, b, c, e, f\}$, or $b\overline{d}a\overline{f}$ corresponding to state $\{a, b, c, e\}$, depending on whether the problem is solved or not. Conversely, truncating the sequence that results from an assessment on Q to the elements of Q_1 gives a sequence that designates a state in $\mathcal{K}_1$. Applying this truncation to the sequences $b\overline{d}a\overline{f}$ and $b\overline{d}af$, for example, gives back $b\overline{d}a$ in both cases.

However, this mutual consistency between local and global assessment does not hold in general. Assume that assessing an individual on the domain Q_2 resulted in the sequence $\overline{a}f\overline{b}$, which corresponds to the state $\{f\}$ in $\mathcal{K}_2$. Then this local assessment cannot be extended to a global assessment on Q. There is no knowledge state in $\mathcal{K}$ that is consistent with this information. The collection of states $\mathcal{K}_{\overline{a}} \cap \mathcal{K}_f \cap \mathcal{K}_{\overline{b}}$ satisfying the constraints in $\overline{a}f\overline{b}$ is empty. Conversely, consider the sequence $ae\overline{f}$ resulting from an assessment on Q which corresponds to the state $\{a, b, c, e\}$ of $\mathcal{K}$. Actually, this sequence may in principle result in an assessment on Q_2, but there is no state in $\mathcal{K}_2$ that corresponds to this sequence. The key to explaining these differences in the relation between local and global assessments is the fact that we have $\mathcal{K}_1 = \mathcal{K}|_{Q_1}$, but $\mathcal{K}_2 \not\subseteq \mathcal{K}|_{Q_2}$ as well as $\mathcal{K}|_{Q_2} \not\subseteq \mathcal{K}_2$ (see Example 1).

The presented results also suggest mechanisms for reconciling skill functions in order to ensure consistency. Proposition 7 offers a simple recipe for such a mechanism. It consists of making the local skill function coincide with the (global) distributed skill function by assigning to each problem all the competencies that are associated to it by any of the skill functions. However, proceeding in this way results in a notion of consistency that is too restrictive. Corollary 2 shows that an appropriate mechanism has to check whether each local skill set can be "covered" by a collection of pairwise compatible skill sets (see Definition 13). It remains to efficiently implement the steps that need to be taken in order to render a collection of skill functions consistent, which is highly relevant for applications. Reconciling the local information in the indicated way, for instance, is useful in both of the above sketched scenarios. In the context of integrating expert queries on overlapping subsets of problems the updating mechanism suggests how to resolve inconsistencies in the resulting skill assignments. A more cautious application is confined to exploiting the fact that the procedure clearly pinpoints the inconsistent skill assignments, which the experts then may reconsider. The quasi-automatical nature of the updating is essential for integrating distributed resources in technology-enhanced learning, especially in open systems, where the local repositories of assessment problems may not be static. In such a setting the consistency of the local skill assignments implemented through metadata tags needs to be established at runtime. Complications may arise in applications like that whenever identical skills appear under different names or labels. Resolving this issue requires defining an equivalence relation on the set of all skills prior to applying the above developed framework.

The results of the present paper also stimulate further questions for which no answers are readily available. These questions come from the fact that the consistency of skill functions is characterized through the meshability of the delineated knowledge structures. This characterization has not yet been fully traced back to properties of the skill functions themselves, although Corollary 2 takes a first step in this direction. The main issue here is that one has to survey the class of all skill functions that delineate a certain knowledge structure. This fixes the domain but leaves us to consider all finite skill sets and the therein possible skill assignments. It remains to identify those structural constraints that characterize classes of skill functions delineating knowledge structures which satisfy relevant properties, such as meshability.

Acknowledgments

We are grateful to two anonymous referees for their valuable comments on a previous draft. The work presented in this paper was supported by the European Community under the Information Society Technologies (IST) program of the 6th FP for RTD-project ELeGI (contract no. IST-002205). The authors are solely responsible for the content of this paper. It does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

Albert, D., Schrepp, M., & Held, T. (1994). Construction of knowledge spaces for problem solving in chess. In G. H. Fischer, & D. Laming (Eds.), *Contributions to mathematical psychology, psychometrics and methodology* (pp. 123–135). New York: Springer.

Conlan, O., Hockemeyer, C., Wade, V., & Albert, D. (2002). Metadata driven approaches to facilitate adaptivity in personalized elearning systems. *The Journal of Information and Systems in Education*, *1*, 38–44.

Doignon, J. (1994). Knowledge spaces and skill assignments. In G. H. Fischer, & D. Laming (Eds.), *Contributions to mathematical psychology, psychometrics and methodology* (pp. 111–121). New York: Springer.

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*. New York: Springer.

Düntsch, I., & Gediga, G. (1995). Skills and knowledge structures. *British Journal of Mathematical and Statistical Psychology*, *48*, 9–27.

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

Falmagne, J.-C., & Doignon, J.-P. (1998). Meshing knowledge structures. In C. E. Dowling, F. S. Roberts, & P. Theuns (Eds.), *Recent progress in mathematical psychology* (pp. 143–153). Hillsdale, USA: Lawrence Erlbaum.

Gediga, G., & Düntsch, I. (2002). Skill set analysis in knowledge structures. *British Journal of Mathematical and Statistical Psychology*, *55*, 361–384.

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

Korossy, K. (1999). Modeling knowledge as competence and performance. In D. Albert, & J. Lukas (Eds.), *Knowledge spaces: Theories, empirical research applications* (pp. 103–132). Mahwah, NJ: Lawrence Erlbaum.

Lukas, J., & Albert, D. (1993). Knowledge assessment based on skill assignment and psychological task analysis. In G. Strube, & K. F. Wender (Eds.), *The cognitive psychology of knowledge* (pp. 139–160). Amsterdam: North-Holland.

Stefanutti, L., Albert, D., & Hockemeyer, C. (2005). Derivation of knowledge structures for distributed learning objects. In P. Ritrovato, C. Allison, S. A. Cerri, T. Dimitrakos, M. Gaeta, & S. Salerno (Eds.), *Towards the learning grid: Advances in human learning services* (pp. 105–112). Amsterdam: IOS Press.