Proceedings of I-KNOW '05
Graz, Austria, June 29 - July 1, 2005

Propositional Knowledge Bases Merging

Z. M. Ma
(College of Information Science and Engineering, Northeastern University
Shenyang, Liaoning 110004, China
zongmin_ma@yahoo.com)

Abstract: This paper concentrates on merging knowledge repositories representing different viewpoints, with all of them subject to updates and revisions. A set of criteria of interest is defined. On the basis, different approaches are reviewed, which are assessed with respect to these criteria.

Keywords: knowledge bases, propositional knowledge, knowledge base merging
Categories: I.2.4

1 Introduction

In the field of artificial intelligence and databases, we are often confronted with multiple sources that need to be reconciled into a single, encompassing repository. The merging of different sources is complicated by potential conflicts and inconsistencies between the different repositories. Extensive research has addressed the problem of merging focusing on various aspects. In database integration, current efforts have put on semantic data model integration and object-oriented database integration [Parent, 98; Pitoura, 95; Palopoli, 00]. More recently, there are some studies that focus on the integrations of behavior [Preuner, 01] and integrity constraint [Turker, 00] in object-oriented databases. It should be noted that, however, in addition to database model based integration, there exists theory based database integration. [Lin, 98] presented a formal approach to merging databases with constraints, where a constraint is a sentence of a function-free first order language. A (first-order) theory is a set of sentences. Relational databases are represented by the theories and they are thereby merged through merging the theories (knowledge).

In knowledge base merging, some combination operators have been proposed [Baral, 91; Baral, 92]. These operators are all based on the union of all the knowledge bases and on the selection of some maximal subsets. Since the problem is to find coherent information from an inconsistent knowledge base once the union of the knowledge bases is done, such a definition of knowledge base combination is very close to Brewka's preferred subtheories [Brewka, 89] and to the work of Benferhat et al. on entailment in inconsistent databases [Benferhat, 93 and 98]. Not defining combination operators for knowledge bases, some researchers have focused on the logical properties that merging operators should satisfy. In [Liberatore, 98], the postulates were proposed to characterize arbitration. Their definition has a strong connection with revision operators [Katsuno, 91]. The major drawback of their definition is that those operators arbitrate only two knowledge bases. Furthermore, they only select some interpretations in the two knowledge bases as the result of the arbitration. Lin and Mendelzon proposed a theory merging by majority operator that solves conflicts between knowledge bases by taking the majority into account [Lin, 99]. In [Konieczny, 98], a set of postulates that a rational merging operator has to satisfy was proposed and arbitration operators and majority operators were distinguished. Here the arbitration operators strive to minimize individual dissatisfaction and the majority operators strive to minimize global dissatisfaction. Furthermore, in [Konieczny, 99 and 02], a logical framework for knowledge base merging in the presence of integrity constraints was presented when there is no preference over the knowledge bases. Also a set of postulates that a rational merging operator with integrity constraints has to satisfy was proposed and arbitration operators and majority operators were distinguished.

Conflict (or inconsistency) problems occur when several expert systems are combined. Halpern and Moses discuss *implicit* conflicts that are only discovered when knowledge is combined [Halpern, 92]. For example, two sources may state respectively $\{a\}$ and $\{a \rightarrow b\}$, both knowing that b is not true; yet, when they sources are combined, they add b as a fact. It is clear that this process allows to discover new piece of belief distributing among the sources.

In this paper, we are interested in merging knowledge repositories representing different viewpoints, with all of them subject to updates and revisions. We define a set of criteria of interest, then review different approaches and assess them with respect to these criteria. The remainder of the paper is organized as follows: In section 2 we introduce background information; in section 3 we define a set of criteria relevant to the merging of knowledge bases; in section 4 we introduce various merging operations and evaluate them with respect to the criteria. In section 5 we conclude this paper.

2 Background Information

2.1 Context

We are interested in maintaining and using knowledge repositories of engineering design knowledge. These repositories contain design constraints on engineering artifacts. For example, the knowledge related to the design of engines may consist of a variety of engine classes with associated collections of validity conditions that we refer to as design constraints. The same repository would contain knowledge about components and of engines and other related parts. These repositories are commonly used to collect information about state of the art in science and technology as well as in-house experience and best-practices. The overall knowledge related to engines -- or any other engineering artifact of some level of complexity—is typically distributed over many areas of specialty.

2.2 Propositional Knowledge Bases

Propositional logic consists in dealing with *prepositional formulas* or *prepositional sentences*, which are formed from a finite number of *propositional variables*, their negations, and the logical connectors. Propositional variables can take only two values: *True* (1) or *False* (0). As there are only two values available, the negation of a variable can be defined as *NOT*, denoted $\neg$.

Using a finite number of propositional variables and their negations as well as the connectors, propositional sentences are formed. In the following, propositional sentences are formally defined.

Definition (*Sentence*): Let $U = <\xi, \mathfrak{R}>$ be a universe of discourse. Here, ξ is the set of propositional variables and $\mathfrak{R}$ is the set of the connectors {$\vee$ (*or*), $\wedge$ (*and*), $\Rightarrow$ (*implication*), $\Leftrightarrow$ (*equivalence*)}. A sentence in *U* is as follows.

(1) For $A \in \xi$, *A* is a sentence;

(2) For $A, B \in \xi$ and $\theta \in \mathfrak{R}$, $(A\ \theta B)$ is a sentence;

(3) For a sentence *S*, $\neg (S)$ is a sentence;

(4) For two sentences *S1* and *S2* and $\theta \in \mathfrak{R}$, $(S1\, \theta\, S2)$ is a sentence.

The main notion in propositional logic is the notion of *valuation*. It consists in assigning a truth value (1/Trure or 0/False) to a given sentence according to truth values given to the variables. It is clear that there are multiple assignments of truth values to the variables but the truth value of the sentence is only 1/Trure or 0/False. Then we have the following definition.

Definition (*Possible worlds and model*): Given a sentence φ formed from *k* variables. A *possible world* or *interpretation* of φ is a mapping from φ to $\{0, 1\}^k$, where a truth value is assigned to each one of *k* variables. A *model* of φ is a possible world where it makes φ true.

For a given sentence, there may be several models. Let φ be a sentence. The set of all the models of φ is denoted by $[[\varphi]]$.

A *propositional knowledge base* is defined as a finite set of the sentences in prepositional logic. In the following, knowledge bases are used to refer to propositional knowledge bases if there is no special indication.

3 Requirements and Criteria for Merging Operations

3.1 Requirements

[Konieczny, 98] underlines the differences between majority operation and arbitration operation in merging and propose a characterization of merging operations that an operation has to satisfy in order to have a rational behavior concerning the merging. Let *E* be a knowledge set, and let Δ be an operation that assigns to each knowledge set *E* to a knowledge base $\Delta(E)$. Then

(M-KPP1) $\Delta(E)$ is consistent

(M-KPP2) If *E* is consistent then $\Delta(E) = \wedge E$

(M-KPP3) If $E1 \leftrightarrow E2$ then $\vdash \Delta(E1) \leftrightarrow \Delta(E2)$

(M-KPP4) If $K \wedge K'$ is not consist then $\Delta(K \cup K') \not\vdash K$

(M-KPP5) $\Delta(E1) \wedge \Delta(E2) \vdash \Delta(E1 \cup E2)$

(M-KPP6) If Δ ($E1$) $\wedge$ Δ ($E2$) is consist then Δ ($E1 \cup E2$) $\vdash$ Δ ($E1$) $\wedge$ Δ ($E2$)

(*M_Maj-KPP7*) $(\forall K)(\exists n)(\Delta(E \cup K^n) \vdash K)$

(*M_Arb-KPP7*) $(\forall K')(\exists K)(K' \nvdash K)(\forall n)(Merge(K' \cup K^n) = Merge(K' \cup K))$

(M-KPP1) shows that the result of merging is always consistent and conflicts among the knowledge bases are always resolved. (M-KPP2) tales care of one limiting case, saying that if there is no conflict among the knowledge bases, then the result of the merging is simply the conjunction of the knowledge bases. (M-KPP3) states that the merging operation is independent of the syntax of each knowledge base, i.e., if two knowledge base sets are equivalent, then the two knowledge bases resulting from the merging are logically equivalent. (M-KPP4) means that when two knowledge bases are merged, merging operations must not give preference to one of them. (M-KPP5) and (M-KPP6) together state that if you could find two subgrounds that agree on at least one alternative, then the result of the global arbitration will be exactly those alternatives the two groups agree on. (*M_Maj-KPP7*) is to say that if an opinion has a large audience, then it will be the opinion of the group. (*M_Arb-KPP7*) states that, to a large extent, the result of the arbitration is independent from the frequency of the different views.

3.2 Criteria and Associated Distances and Metrics

A merging operation for knowledge bases must satisfy the requirements given above. But there still are many possible merging operations that all satisfy the given requirements and it is unclear how we can distinguish or choose between them. For this purpose, let us focus on the example of merging operation given in [Konieczny, 98]. But first of all, we present the Dalal's distance [Dalal, 88] to calculate the distance between two possible worlds $w1$ and $w2$, denoted $dist(w1, w2)$. Then $dist(w1, w2)$ is the number of the variables that take different values in $w1$ and $w2$.

Extending this definition, the distance between a possible world w and a sentence φ, denoted $dist(w, \varphi)$, can be defined as a minimum of the distances between w and all worlds of *the given sentence*. Formally, $dist(w, \varphi)$ is defined as

$$dist(w, \varphi) = \min_{w' \in [[\varphi]]} (dist(w, w')).$$

Finally, the distance between two sentences φ and φ', denoted $dist(\varphi, \varphi')$, can be defined as follows.

$$dist(\varphi, \varphi') = \min_{w \in [[\varphi]] \text{ and } w' \in [[\varphi']]} (dist(w, w'))$$

In [Konieczny, 98], a definition of distance between an interpretation and a set of knowledge bases, $dist_\Sigma$, is given, which is the sum of the distances between this interpretation and the knowledge bases of the set. Formally, let E be a set of knowledge bases and let I be an interpretation. Then the distance between E and I is defined as

$$dist_\Sigma(I, E) = \sum_{\varphi \in E} dist(I, \varphi).$$

Then the following order is defined as

I $\leq_E^{\Sigma} \leq$ J iff $dist_{\Sigma}$ $(I, E) \leq dist_{\Sigma}$ (I, E),
where *I* and *J* are two different interpretations.

Based on this order, a merging operation Δ_{Σ} (majority) is defined as follows.

$[[\Delta_{\Sigma} (E)]] = \min (\leq_E^{\Sigma})$

That means that we always look for such interpretations in the universe of discourse that has the minimum distance to the set of knowledge bases. It has been proven in [Konieczny, 98] that the merging operation Δ_{Σ} satisfies (M-KPP1)-(M-KPP6) and (*M_Maj-KPP7*).

Consider the example given in [Konieczny, 98]. Let a set of knowledge bases *class* with three students $E = \{\varphi 1, \varphi 2, \varphi 3\}$. The teacher can teach SQL, Datalog and O2, which are represented by *S*, *D* and *O*, respectively. He asks these three students in turn to choose what to teach to satisfy the class best.

- The first student wants to learn SQL or O2: $\varphi 1 = (S \vee O) \wedge \neg D$
- The second student wants to learn Datalog or O2 but not both: $\varphi 2 = (\neg S \wedge D \wedge \neg O) \vee (\neg S \wedge \neg D \wedge O)$
- The third student wants to learn three languages: $\varphi 3 = (S \wedge D \wedge O)$

Then we have

- $[[\varphi 1]] = \{(1, 0, 0), (0, 0, 1), (1, 0, 1)\}$
- $[[\varphi 2]] = \{(0, 1, 0), (0, 0, 1)\}$
- $[[\varphi 3]] = \{(1, 1, 1)\}$

All distances between the possible worlds and knowledge bases as well as the set of knowledge bases are shown in Table1.

Possible worlds I	**$\varphi 1$**				**$\varphi 2$**			**$\varphi 3$**		***$dist_{\Sigma}$***
	[[$\varphi 1$]]			***dist (I, $\varphi 1$)***	**[[$\varphi 2$]]**		***Dist (I, $\varphi 2$)***	**[[$\varphi 3$]]**	***Dist (I, $\varphi 3$)***	
	(1, 0, 0)	**(0, 0, 1)**	**(1, 0, 1)**		**(0, 1, 0)**	**(0, 0, 1)**		**(1, 1, 1)**		
(0, 0, 0)	1	1	2	*1*	1	1	*1*	3	*3*	5
(0, 0, 1)	2	0	1	*0*	2	0	*0*	2	*2*	2
(0, 1, 0)	2	2	3	*2*	0	2	*0*	2	*2*	4
(0, 1, 1)	3	1	2	*1*	1	1	*1*	1	*1*	3
(1, 0, 0)	0	2	1	*0*	2	2	*2*	2	*2*	4
(1, 0, 1)	1	1	0	*0*	3	1	*1*	1	*1*	2
(1, 1, 0)	1	3	2	*1*	1	3	*1*	1	*1*	3
(1, 1, 1)	2	2	1	*1*	2	2	*2*	0	*0*	3

Table 1: Distances

According to the merging operation given above, it is clear that

$[[\Delta_{\Sigma} (E)]] = \{(0, 0, 1), (1, 0, 1)\}$.

The teacher hereby has to teach both SQL and O2 or O2 alone to best fit the class. Formally, we have

$\Delta_{\Sigma} (\varphi 1, \varphi 2, \varphi 3) = (\neg S \wedge \neg D \wedge O) \vee (S \wedge \neg D \wedge O)$.

For this purpose, we introduce some criteria of interests in this section. First we give three major criteria.

- *Traceability*. A merged knowledge base *KB* is traceable iff it can be traced back to original knowledge bases. *Minimality*. A merged knowledge base *KB* is minimal
- *Timeliness*. A merged knowledge base *KB* is timely iff it gives priority to more recent information.

In addition to these three criteria above, we also have two minor criteria as follows.

- *Lack of redundancy*.
- *Ease of update*. Information entered later has higher priority over information entered earlier.

4 Merging Knowledge Bases

4.1 Syntactic Merging Relationship Between Knowledge Bases

Since knowledge bases consist of sentences, semantic relationship between sentences and knowledge bases must be investigated in order to discuss semantic relationship between knowledge bases. We identify one kind of semantic relationship between a sentence and a knowledge base, which is semantic implication.

Definition (*Sentence implication*): Given a knowledge base *KB* and a sentence φ. If *KB* encompasses φ, we say *KB* implies φ, denoted $KB2 \Rightarrow \varphi$.

Based on the sentence implication, now we focus on the semantic relationship between knowledge bases that consist of sentences. We identify two kinds of semantic relationship between knowledge bases: *semantic inclusion* and *semantic equivalence*.

Definition (*Knowledge base inclusion*): Let *KB*1 and *KB*2 be two knowledge bases. Then *KB*2 semantically includes *KB*1, denoted $KB1 \subseteq KB2$, if every sentence in *KB*1 is implied by *KB*2. Formally,

$(\forall \varphi)\ (\varphi \in KB1 \wedge KB2 \Rightarrow \varphi)$.

It is clear that given two knowledge bases *KB*1 and *KB*2, knowledge base $KB = KB1 \cup KB2$ satisfies $KB1 \subseteq KB$ and $KB2 \subseteq KB$, or $KB = KB1 \wedge KB2$ satisfies $KB1 \subseteq KB$ and $KB2 \subseteq KB$ if there is no conflict between *KB*1 and *KB*2.

Definition (*Knowledge base equivalence*): If two knowledge bases *KB*1 and *KB*2 satisfy $KB1 \subseteq KB2$ and $KB2 \subseteq KB1$, *KB*1 and *KB*2 are semantically equivalent to each other, denoted $KB1 = KB2$.

Based on the semantic inclusion relationship between knowledge bases, we can define an ordering relation $\preccurlyeq$ over knowledge bases. The ordering relation $\preccurlyeq$ above can also be used over knowledge bases. Given two knowledge bases *KB*1 and *KB*2, $KB1 \preccurlyeq KB2$ if and only if $KB1 \subseteq KB2$. For $KB1 \preccurlyeq KB2$, *KB*2 provides more information than *KB*1. The order relation $\preccurlyeq$ is

- reflexive, i.e., $\varphi1 \preccurlyeq \varphi1$, and
- transitive, i.e., if $\varphi1 \preccurlyeq \varphi2$ and $\varphi2 \preccurlyeq \varphi3$, then $\varphi1 \preccurlyeq \varphi3$.

Using the ordering relation defined, in the following, we formulate knowledge base merging.

4.2 Merging Knowledge Bases

Intuitively merging knowledge bases is to find a knowledge base that has at least as much information as each component knowledge base and it is the smallest such knowledge base. So we define merging as follows.

Definition. Given three knowledge bases *KB*1, *KB*2, and *KB*3, merging *KB*1, *KB*2, and *KB*3 consists of creating a knowledge base *KB*-if exists-such as:

(1) $KB1 \preccurlyeq KB$, $KB2 \preccurlyeq KB$, and $KB3 \preccurlyeq KB$.

(2) *KB* is the smallest knowledge base.

This definition means that when it exists, *KB* should be the *least upper bound* (*lub*) of *KB*1, *KB*2, and *KB*3. Formally,

KB = Merge (*KB*1, *KB*2, *KB*3) = *lub* (*KB*1, *KB*2, *KB*3)

It should be noticed, however, when we define the merge of several knowledge bases as the least upper bound, we always assume that there exists such the merged knowledge base. When all component knowledge bases are consistent, we have

KB = Merge (*KB*1, *KB*2, *KB*3) = *lub* (*KB*1, *KB*2, *KB*3) = $KB1 \wedge KB2 \wedge KB3$.

At this moment, it is clear that

- Merge (*KB*1, *KB*2, *KB*3) is consistent and
- Merge (*KB*1, *KB*2, *KB*3) = $KB1 \wedge KB2 \wedge KB3$ if *KB*1, *KB*2, and *KB*3 are consistent.

Also it is clear that

- If *KB*1 = *KB*1', *KB*2 = *KB*2', and *KB*3 = *KB*3', then Merge (*KB*1, *KB*2, *KB*3) = Merge (*KB*1', *KB*2', *KB*3').

These three properties are the same as the postulates of knowledge base merging given in [Lin, 99].

However, the assumption that all component knowledge bases are consistent is not always true. When component knowledge bases are not consistent, we have to take

KB = Merge (*KB*1, *KB*2, *KB*3) = *lub* (*KB*1, *KB*2, *KB*3) = $KB1 \cup KB2 \cup KB3$.

Although there exists *lub* (*KB*1, *KB*2, *KB*3) at this moment, it is clear that *lub* (*KB*1, *KB*2, *KB*3) is not consistent. Consider that we want to merge the following knowledge bases: $KB1 = KB2 = \{a, b\}$, $KB3 = \{a, b \rightarrow c\}$, $KB4 = \{\neg a, d\}$. With the merging operation above, *KB* will be $\{a, \neg a, b, b \rightarrow c, d\}$. Since *KB* should be the smallest knowledge base, redundant information is removed from *KB*. It is clear *KB* is not consistent because a and $\neg a$ exist simultaneously.

An inconsistent knowledge base gives no information for the merging process. In this case, merging knowledge bases is defined to *find a knowledge base that is closest to the component knowledge bases instead of having at least as much information as each component knowledge base* and *it is consistent and the smallest such knowledge base*. So the revised definition of merging knowledge bases is defined as follows.

Definition. Given three knowledge bases *KB*1, *KB*2, and *KB*3, merging *KB*1, *KB*2, and *KB*3 consists of creating a knowledge base *KB*-if exists-such as:

(1) *KB* is consistent,

(2) *KB* is closest to {*KB*1, *KB*2, *KB*3}, and

(3) *KB* is the smallest knowledge base.

It is clear that when *KB*1, *KB*2, and *KB*3 are consistent, this definition becomes to the last definition. Now let us focus on the situation that *KB*1, *KB*2, and *KB*3 are not consistent. In this case, we first obtain several maximal consistent subsets of *lub* (*KB*1, *KB*2, *KB*3). Then we select such subset(s) that should have the closest distance with {*KB*1, *KB*2, *KB*3}. According to the different computation of distance between a knowledge base and a set of knowledge bases, we at least have two kinds of merging: *majority* and *arbitration* when *lub* (*KB*1, *KB*2, *KB*3) is not consistent.

First of all, we define the distance between two knowledge bases. The distance between two knowledge bases is the number of the sentences that are not equivalent. Let *KB* and *KB*' be two knowledge bases. The distance between *KB* and *KB*' is denoted by *dist* (*KB, KB*'). Based on the distance between two knowledge bases, now we can define the distance between a knowledge base a set of knowledge bases. Depending on different merging policies (e.g., majority or arbitration), the following definitions for the distance between a knowledge base and a set of knowledge bases can be identified.

Let *KB* be a knowledge base and {*KB*1, *KB*2 , …, *KB*n} be a set of knowledge bases. The distance between *KB* and {*KB*1, *KB*2 , …, *KB*n} is denoted by *dist* (*KB*, {*KB*1, *KB*2 , …, *KB*n}). Then we have

Definition. The distance between *KB* and {*KB*1, *KB*2 , …, *KB*n} is the sum of the distances between *KB* and each knowledge base in {*KB*1, *KB*2 , …, *KB*n}. Formally,

$$dist\ (KB, \{KB1, KB2, \ldots, KBn\}) = \sum_{i=1}^{n} (dist\ (KB, KBi)).$$

This definition is for majority merging operation.

Definition. The distance between *KB* and {*KB*1, *KB*2, …, *KB*n} is a sequence of the distances between *KB* and each knowledge base in {*KB*1, *KB*2 , …, *KB*n}, in which these distances are sorted in descending order.

This definition is for arbitration merging operation.

Again consider three knowledge bases *KB*1 = *KB*2 = {a, b}, *KB*3 = {a, $b \rightarrow c$}, and *KB*4 = {$\neg a$, d}. It is clear that *lub* (*KB*1, *KB*2, *KB*3, *KB*4) = {a, $\neg a$, b, $b \rightarrow c$, d} is not consist. Then we have the following maximal consistent subsets of *lub* (*KB*1, *KB*2, *KB*3, *KB*4): {a, b, $b \rightarrow c$, d} and {$\neg a$, b, $b \rightarrow c$, d}. Each of these two subsets is consistent and the smallest knowledge base (no redundant information). Now we calculate the distance between each of these two subsets and {*KB*1, *KB*2, *KB*3, *KB*4} based on majority and arbitration.

For majority:

dist ({a, b, $b \rightarrow c$, d}, {*KB*1, *KB*2, *KB*3, *KB*4}) = 2 + 2 + 2 + 3 = 9

dist ({$\neg a$, b, $b \rightarrow c$, d}, {*KB*1, *KB*2, *KB*3, *KB*4}) = 4 + 4 + 4 + 2 = 14

It is clear that the distance between {a, b, $b \rightarrow c$, d} and {*KB*1, *KB*2, *KB*3, *KB*4} is less than the distance between {$\neg a$, b, $b \rightarrow c$, d} and {*KB*1, *KB*2, *KB*3, *KB*4} using majority. So

KB = Merge (*KB*1, *KB*2, *KB*3) = {a, b, $b \rightarrow c$, d}.

For arbitration:

dist ({a, b, $b \rightarrow c$, d}, {*KB*1, *KB*2, *KB*3, *KB*4}) = (3, 2, 2, 2)

dist ({$\neg a$, b, $b \rightarrow c$, d}, {*KB*1, *KB*2, *KB*3, *KB*4}) = (4, 4, 4, 2)

It is clear that the distance between {a, b, $b \rightarrow c$, d} and {*KB*1, *KB*2, *KB*3, *KB*4} is less than the distance between {$\neg a$, b, $b \rightarrow c$, d} and {*KB*1, *KB*2, *KB*3, *KB*4} using arbitration. So

KB = Merge (*KB*1, *KB*2, *KB*3) = {a, b, $b \rightarrow c$, d}.

Based on the discussions above, we now give a simple algorithm for merging knowledge bases.

Algorithm *Merge* (*KB*1, *KB*2 , …, *KB*n)

Input: a set of knowledge bases {*KB*1, *KB*2 , …, *KB*n}

Output: *Merge* (*KB*1, *KB*2 , …, *KB*n)

Step 1: Obtain *lub* (*KB*1, *KB*2 , …, *KB*n) = *KB*1 ∪ *KB*2 ∪ …∪ *KB*n;

Step 2: Determine if *lub* (*KB*1, *KB*2 , …, *KB*n) is consistent;

Step 3: If *lub* (*KB*1, *KB*2 , …, *KB*n) is consistent, then *Merge* (*KB*1, *KB*2 , …, *KB*n) = *KB*1 ∧ *KB*2 ∧ …∧ *KB*n and goto Step 8;

Step 4: Partition *lub* (*KB*1, *KB*2 , …, *KB*n) into several maximal consistent subsets;

Step 5: Determine merging policy;

Step 6: For each partition *T*, compute *dist* (*T*, {*KB*1, *KB*2 , …, *KB*n});

Step 7: Select such *T* that has the minimum distance to {*KB*1, *KB*2 , …, *KB*n} and *Merge* (*KB*1, *KB*2 , …, *KB*n) = *Merge* (*KB*1, *KB*2 , …, *KB*n) + *T*;

Step 8 Return (*Merge* (*KB*1, *KB*2 , …, *KB*n))

4.3 Properties of the Merging Operation

Now let us focus on how the merging operation defined above satisfies that the postulates of knowledge base merging given in [Konieczny, 98]. It is not difficult to see the merging operation defined in the paper satisfies (M-KPP1)-(M-KPP3) according to the definition of the merging operation. So we omit their proofs. Now let us look at if (M-KPP4) is satisfiable for the merging operation. Assume that $K \wedge K'$ is not consistent completely. That means there is no such a sentence in one knowledge base that is consistent with any sentence in another knowledge base. In other words, we can only get two maximal consistent subsets K and K' from inconsistent $K \wedge K'$. By the definition of the merging operation, we have

$\Delta\ (K \cup K') = K$ if $dist\ (K, \{K, K'\}) < dist\ (K', \{K, K'\})$ or

$\Delta\ (K \cup K') = K'$ if $dist\ (K, \{K, K'\}) > dist\ (K', \{K, K'\})$.

So $\Delta\ (K \cup K') \not\vdash K$ holds.

Consider an example. Let K1 = {a → c, e → ¬c, b → ¬c}, K2 = {a, e}, and K3 = {b}. Then Merge (K1, K2) ∧ Merge (K3) is either

{a, a → c, e → ¬c, b → ¬c} ∧ {b} or

{e, a → c, e → ¬c, b → ¬c} ∧ {b},

depending on which distances of {a, a → c, e → ¬c, b → ¬c} and {e, a → c, e → ¬c, b → ¬c} to {K1, K2} is smaller. It can be seen that Merge (K1, K2) ∧ Merge (K3) is consistent. However, Merge (K1, K2, K3) should be one of followings

{a, a → c, e → ¬c, b},

{a, a → c, e → ¬c, b → ¬c},

{e, a → c, e → ¬c, b}, or

{e, a → c, e → ¬c, b → ¬c}.

Merge (K1, K2, K3) $\not\vdash$ Merge (K1, K2) ∧ Merge (K3). (M-KPP6) does not hold.

5 Conclusions

This paper concentrated on merging knowledge repositories representing different viewpoints, with all of them subject to updates and revisions. A set of criteria of interest was proposed. On the basis, different approaches were reviewed, which were assessed with respect to these criteria.

References

[Baral, 91] Baral, C., Kraus, S. and Minker, J., 1991, Combining Multiple Knowledge Bases, *IEEE Transactions on Knowledge and Data Engineering*, 3 (2): 208-220.

[Baral, 92] Baral, C., Kraus, S., Minker, J. and Subrahmanian, V. S., 1992, Combining Knowledge Bases Consisting of First-Order Theories, *Computational Intelligence*, 8 (1): 45-71.

[Benferhat, 93] Benferhat, S., Cayrol, C., Dubois, D., Lang, J. and Prade, H., 1993, Inconsistency Management and Prioritized Syntax-Based Entailment, *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 640-647.

[Benferhat, 98] Benferhat, S., Dubois, D., Lang, J., Prade, H., Saffiotti, A. and Smets, P., 1998, A General Approach for Inconsistency Handling and Merging Information in Prioritized Knowledge Bases, *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning*, 466-477.

[Brewka, 89] Brewka, G., 1989, Preferred Subtheories: An Extended Logical Framework for Default Reasoning, *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, 1043-1048.

[Dalal, 88]Dalal, M., 1988, Investigation into a Theory of Knowledge Base Revision: Preliminary Report, *Proceedings of the Seventh National Conference of the American Association for Artificial Intelligence*, 475-479.

[Halpern, 92] Halpern, J. Y. and Moses, Y., 1992, A Guide to Completeness and Complexity for Modal Logics of Knowledge and Belief, *Artificial Intelligence*, 54 (2): 319-379.

[Katsuno, 91] Katsuno, H. and Mendelzon, A. O., 1991, On the Difference Between Updating a Knowledge Base and Revising It, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, 387-394.

[Konieczny, 98] Konieczny, S. and Pino-Pérez, R., 1998, On the Logic of Merging, *Proceedings of Sixth International Conference on Principles of Knowledge Representation and Reasoning*, 488-498.

[Konieczny, 99] Konieczny, S. and Pino-Pérez, R., 1999, Merging with Integrity Constraints, *Proceedings of Fifth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, LNCS 1638, 233-244.

[Konieczny, 02] Konieczny, S. and Pino-Pérez, R., 2002, Merging Information under Constraints: A Logic Framework, *Journal of Logic and Computation*, 12 (5), 773-808.

[Liberatore, 98] Liberatore, P. and Schaerf, M., 1998, Arbitration (or How to Merge Knowledge Bases), *IEEE Transactions on Knowledge and Data Engineering*, 10 (1): 76 -90.

[Lin, 98] Lin, J. X. and Mendelzon, A. O., 1998, Merging Databases under Constraints, *International Journal of Cooperative Information Systems*, 7 (1): 55-76.

[Lin, 99] Lin, J. X. and Mendelzon, A. O., 1999, Knowledge Base Merging by Majority, in R. Pareschi and B. Fronhoefer (eds.), *Dynamic Worlds: From the Frame Problem to Knowledge Management*, Kluwer.

[Mili, 00] Mili, F., 2000, Managing Engineering Knowledge Federations, *Industrial Knowledge Management – A Micro Level Approach* (Edited by R. Roy), Springer Verlag, London, 513-524.

[Palopoli, 00] Palopoli, L., Pontieri, L., and Terracina, G. et al., 2000, Intensional and extensional integration and abstraction of heterogeneous databases, *Data & Knowledge Engineering*, 35 (3): 201-237.

[Parent, 98] Parent, C. and Spaccapietra, S., 1998, Issues and approaches of database integration, *Communications of the ACM*, 41 (5): 166-178.

[Pitoura, 95] Pitoura, E., Bukhres, O. A. and Elmagarmid, A. K., 1995, Object orientation in multidatabase systems, *ACM Computing Surveys*, 27 (2): 141-195.

[Preuner, 01] Preuner, G., Conrad, S. and Schrefl, M., 2001, View integration of behavior in object-oriented databases, *Data & Knowledge Engineering*, 36 (2): 153-183.

[Turker, 00] Turker, C. and Saake, G., 2000, Global extensional assertions and local integrity constraints in federated schemata, *Information Systems*, 25 (8): 503-526.