Syntactic Propositional Belief Bases Fusion with Removed Sets

Julien Hue[1], Odile Papini[2], and Eric Wurbel[1]

[1] LSIS-CNRS 6168, Université du Sud Toulon -Var. BP 132. 83957 La Garde Cedex France
{hue, wurbel}@univ-tln.fr
[2] LSIS-CNRS 6168, ESIL, Université de la Méditerranée. Av de Luminy. Case 13009 Marseille Cedex 9 France
papini@esil.univmed.fr

Abstract. The problem of merging multiple sources information is central in several domains of computer science. In knowledge representation for artificial intelligence, several approaches have been proposed for propositional bases fusion, however, most of them are defined at a semantic level and are untractable. This paper proposes a new syntactic approach of belief bases fusion, called Removed Sets Fusion (RSF). The notion of removed-set, initially defined in the context of belief revision is extended to fusion and most of the classical fusion operations are syntactically captured by RSF. In order to efficiently implement RSF, the paper shows how RSF can be encoded into a logic program with answer set semantics, then presents an adaptation of the smodels system devoted to efficiently compute the removed sets in order to perform RSF. Finally a preliminary experimental study shows that the answer set programming approach seems promising for performing belief bases fusion on real scale applications.

1 Introduction

Merging information coming from different sources is an important issue in various domains of computer science like knowledge representation for artificial intelligence, decision making or databases. The aim of fusion is to obtain a global point of view, exploiting the complementarity between sources, solving different existing conflicts, reducing the possible redundancies. Among the various approaches of multiple sources information merging, logical approaches gave rise to increasing interest in the last decade [1,2,3,4,5]. Most of these approaches have been defined within the framework of classical logic, more often propositional, and have been semantically defined. Different postulates characterizing the rational behavior of fusion operators have been proposed [6] and various operators have been defined according to whether explicit or implicit priorities are available [6], [7], [8], [9], [10], [11]. More recently, new approaches have been proposed like semantic merging for propositional bases, stemming from the Hamming distance [12] or syntactic fusion in a possibilistic framework [13,14] which is a real advantage at a computational point of view.

K. Mellouli (Ed.): ECSQARU 2007, LNAI 4724, pp. 66–77, 2007.

This paper proposes a new approach for performing syntactic fusion of propositional belief bases. We show that the classical fusion operations $Card$, Σ, Max, $GMax$, initially defined at the semantic level, can be expressed within our syntactic framework. We then show that an efficient implementation of these operations, based on answer set programming, can be performed. In particular this paper focuses on the following three issues:

– We extend the Removed Sets Revision to the fusion of propositional belief bases, called Removed Sets Fusion (RSF). We show how the notion of removed-set, roughly speaking, the subsets of clauses to remove to restore consistency, initially defined in the context of belief bases revision [15,16] is generalized to the case of belief bases fusion. We then show that classical fusion operations are captured within this framework since each fusion strategy is encoded by a preference relation between subsets of clauses.
– In the last decade, answer set programming has been considered as a convenient tool to handle non-monotonic reasoning systems. Moreover, several efficient systems have been developed [17], [18], [19], [20], [21]. We propose to formalize the Removed Sets Fusion in terms of answer set programming and to adapt the smodels system in order to compute preferred answer sets which correspond to removed sets. Therefore, to propose an effective computational fusion algorithm.
– The conducted preliminary experimental study illustrates the behaviour of RSF for the $Card$, Σ strategies and seems promising for performing fusion in real scale applications.

The rest of this paper is organized as follows. The next section fixes the notations and gives a refresher on fusion, removed set revision and on answer set programming. The paper then presents the Removed Set Fusion. It shows how Removed Sets Fusion is encoded into logic programming with answer set semantics and presents an adaptation of the Smodels system for computing answer sets and performing Removed Sets Fusion. It then presents a preliminary experimental study which illustrates the approach and shows that the answer set programming implementation seems promising before concluding.

2 Background and Notations

We consider a propositional language $\mathcal{L}$ over a finite alphabet $\mathcal{P}$ of atoms. A literal is an atom or the negation of an atom. The usual propositional connectives are denoted by $\neg$, $\wedge$, $\vee$ and Cn denotes the logical consequence. A *belief base* K is a finite set of propositional formulae over a propositional language $\mathcal{L}$.

2.1 Fusion

Let $E = \{K_1, \ldots, K_n\}$ be a multi-set of n consistent belief bases to be merged, E is called a *belief profile*. The n belief bases $K_1, \ldots, K_n$ are not necessarily different and the union of belief bases, taking repetitions into account, is denoted by $\sqcup$

and their conjunction and disjunction are denoted by $\bigwedge$ and $\bigvee$ respectively. For the sake of simplicity, we denote by K the belief set consisting of the singleton $E = \{K\}$.

We define a fusion operator Δ as a function which associates to each belief profile a classical consistent belief base denoted by $\Delta(E)$. In the literature, there are two different ways for defining $\Delta(E)$: either using some implicit priority or not. In the following implicit priority is not assumed.

There are two straighforward ways for defining $\Delta(E)$ depending if the sources are conflicting or not, the classical conjunctive merging : $\Delta(E) = \bigwedge_{K_i \in E} K_i$ suitable when the sources are not conflicting and the classical disjunctive merging : $\Delta(E) = \bigvee_{K_i \in E} K_i$ appropriate in case of conflicting sources. These two opposite cases are not satisfactory, then several methods have been proposed for fusion according to whether the bases have the same importance or not.

In particular, the following classical fusion operators have been proposed. The *Cardinality operator*, denoted by $Card$, [1] which takes the number of the belief bases of E into account. The *Sum operator*, denoted by Σ, [22,2]which follows the point of view of the majority of the belief bases of E.

The *Max-based* operator, denoted by Max [4], which tries to satisfy all the belief bases of E. The *Leximax-based* operator, denoted by $GMax$, [6] which tries to satisfy all the belief bases of E, taking the belief bases into account, according to a lexicographic ordering over them.

Different postulates characterizing the rational behaviour of fusion operators have been proposed [6]. Moreover, the various operators have been classified according to two families: the majority and the arbitration ones.

2.2 Answer Sets

A *normal logic program* is a set of rules of the form $c \leftarrow a_1, ..., a_n, \mathit{not}\ b_1, ..., \mathit{not}\ b_m$ where $c, a_i (1 \leq i \leq n), b_j (1 \leq j \leq m)$ are propositional atoms and the symbol *not* stands for *negation as failure*. For a rule r like above, we introduce $\mathit{head}(r) = c$ and $\mathit{body}(r) = \{a_1, ..., a_n,\ b_1, ..., b_m\ \}$. Furthermore, let $\mathit{body}^+(r) = \{a_1, ..., a_n\}$ denotes the set of positive body atoms and $\mathit{body}^-(r) = \{b_1, ..., b_m\}$ the set of negative body atoms, and $\mathit{body}(r) = \mathit{body}^+(r) \cup \mathit{body}^-(r)$.

Let r be a rule, r^+ denotes the rule $\mathit{head}(r) \leftarrow \mathit{body}^+(r)$, obtained from r by deleting all negative body atoms in the body of r.

A set of atoms X is *closed under* a basic program P iff for any rule $r \in P$, $\mathit{head}(r) \in X$ whenever $\mathit{body}(r) \subseteq X$. The smallest set of atoms which is closed under a basic program P is denoted by $CN(P)$.

The *reduct* or Gelfond-Lifschitz transformation [23], P^X of a program P relatively to a set X of atoms is defined by $P^X = \{r^+ \mid r \in P \mathit{\ and\ } \mathit{body}^-(r) \cap X = \emptyset\}$.

A set of atoms X is an *answer set* of P iff $CN(P^X) = X$.

Definition 1. *Let L be a set of literals and A be a set of atoms. L covers A iff $A \subseteq \mathit{Atom}(L)$.*

2.3 Smodels

Smodels is the first and simplest answer set solver [24]. It's a Branch and Bound algorithm (see Algorithm 1) that builds, as one goes along, a set of atoms A representing a potential answer set. It uses the following functions: $expand(A)$ which computes the immediate consequences of A, $conflict(A)$ which detects the conflicts that may arise after the expansion and $heuristic(A)$ which tries to reduce the search space by maximizing the number of deduced atoms. The function $heuristic(A)$ amounts to reduce the number of next atoms to select and makes the conflicts detection faster.

Algorithm 1. $smodels(A)$

```
A ← expand(A)
if conflict(A) then
  return false
else if A covers atom(E) then
  return true
else
  x ← heuristic(A)
  if smodels(A ∪ {x}) then
    return true
  else
    return smodels(A ∪ {not x})
  end if
end if
```

2.4 Removed Sets Revision

We briefly recall the Removed Sets Revision approach. The Removed Sets Revision [16] deals with the revision of a set of propositional formulae by a set of propositional formulae [1]. Let K and A be finite sets of clauses. The Removed Sets Revision focuses on the minimal subsets of clauses to remove from K, called *removed sets* [15], in order to restore the consistency of $K \cup A$. More formally:

Definition 2. *Let K and A be two consistent sets of clauses such that $K \cup A$ is inconsistent. R a subset of clauses of K, is a* removed set *of $K \cup A$ iff (i) $(K \backslash R) \cup A$ is consistent; (ii) $\forall R' \subseteq K$, if $(K \backslash R') \cup A$ is consistent then $\mid R \mid \leq \mid R' \mid$[2].*

Let denote by $\mathcal{R}(K \cup A)$ the collection of removed sets of $K \cup A$, the Removed Sets Revision (RSR) is defined as follows:

Definition 3. *Let K and A be two consistent sets of clauses. The removed sets revision is defined by: $K \circ_{RSR} A =_{def} \bigvee_{R \in \mathcal{R}(K \cup A)} Cn((K \backslash R) \cup A)$.*

[1] From now on, we consider propositional formulae in their equivalent conjunctive normal form (CNF).

[2] $\mid R \mid$ denotes the number of clauses of R.

3 Removed Sets Fusion

We propose a new syntactic fusion framework, Removed Set Fusion (RSF), which aims at merging several consistent belief bases. The approach consists in removing subsets of clauses from the union of the belief bases, according to a given strategy P in order to restore consistency. This framework captures the classical fusion operators and can be efficiently implemented. It generalizes the previously recalled RSR belief revision operation and requires a generalization of the notion of removed set.

Let $E = \{K_1, \ldots, K_n\}$ be a belief profile where K_i, $1 \leq i \leq n$ is a consistent belief base and let X and X' be two subsets of $K_1 \sqcup \ldots \sqcup K_n$.

Definition 4. *Let $E = \{K_1, \ldots, K_n\}$ be a belief profile such that $K_1 \sqcup \ldots \sqcup K_n$ is inconsistent, $X \subseteq K_1 \sqcup \ldots \sqcup K_n$ is a* potential removed set *of E iff $(K_1 \sqcup \ldots \sqcup K_n) \backslash X$ is consistent.*

The number of the potential removed sets is exponential with respect to the number of clauses in E. Hence, only the most relevant potential removed sets, according to a the chosen strategy, have to be selected. Therefore, a preference relation according to any strategy P, denoted by $\leq_P$, is defined and $X \leq_P X'$ means that X is preferred to X' according to P.

Definition 5. *Let $E = \{K_1, \ldots, K_n\}$ be a belief profile such that $K_1 \sqcup \ldots \sqcup K_n$ is inconsistent, $X \subseteq K_1 \sqcup \ldots \sqcup K_n$ is a* removed set *of E according to P iff*

1. *X is a potential removed set of E;*
2. *There is no $X' \subseteq K_1 \sqcup \ldots \sqcup K_n$ such that $X' <_P X$.*

We denote by $\mathcal{F}_P\mathcal{R}(E)$ the collection of removed sets [3] of E according to P, the Removed Set Fusion (RSF) is defined as follows.

Definition 6. *Let $E = \{K_1, \ldots, K_n\}$ be a belief profile. The fusion operation $\Delta^P(E)$ is defined by:*

$$\Delta^P(E) = \bigvee_{X \in \mathcal{F}_P\mathcal{R}(E)} \{Cn((K_1 \sqcup \ldots \sqcup K_n) \backslash X)\}$$

Classical merging operators are easy to use in this context, by instanciating the preceding definitions with the preference relations defined next.

3.1 Representing Classical Fusion Operations with RSF

We here define some of the classical merging operators ($Card$,Σ, Max, $Gmax$,...). They can be encoded by preference relations over potential removed sets.

[3] If $K_1 \sqcup \ldots \sqcup K_n$ is consistent $\mathcal{F}_P\mathcal{R}(E) = \emptyset$.

***Card* operation.** The $Card$ operation is captured within our framework as follows:

Definition 7. *Let X and X' be two potential removed sets of E without repetition: $X \leq_{Card} X'$ iff $|X| \leq |X'|$.*

The $Card$ strategy minimizes the number of clauses to remove from E and does not take repetitions into account. It is close to the $Comb4$ operator defined in [1].

Σ operation. The Σ operation is captured within our framework as follows:

Definition 8. *Let X and X' be two potential removed sets of E: $X \leq_{\Sigma} X'$ iff $\sum_{1 \leq i \leq n} |X \cap K_i| \leq \sum_{1 \leq i \leq n} |X' \cap K_i|$.*

The Σ strategy minimizes the number of clauses to remove from E taking repetitions into account. It corresponds to the *intersection operator* defined in [25].

***Max* operation.** The Max operation is captured within our framework as follows:

Definition 9. *Let X and X' be two potential removed sets of E: $X \leq_{max} X'$ iff*
$\max_{1 \leq i \leq n} |X \cap K_i| \leq \max_{1 \leq i \leq n} |X' \cap K_i|$ and $X \subseteq X'$.

The Max strategy tries to spread the clauses to remove over the belief bases of E and minimizes the number of clauses to remove from the belief base the most involved in the inconsistency.

***Gmax* operation.** The $Gmax$ operation is captured within our framework as follows:

Definition 10. *For each potential removed sets X and each belief base K_i, we define $p^i_X = |X \cap K_i|$. Let L^E_X be the sequence $(p^1_X, \ldots, p^n_X)$ given in a decreasing order. Let X and X' be two potential removed sets of E: $X \leq_{Gmax} X'$ iff $L^E_X <_{lex} L^E_{X'}$ [4].*

The $GMax$ strategy is a refinement of the Max strategy it removes clauses from the belief bases according to a decreasing order on the number of clauses involved in the inconsistency.

Example. We use the following example [2] to illustrate our framework. Consider the following situation : a teacher asks to his students which among the following languages SQL (denoted by s), O_2 (denoted by o) and Datalog (denoted by d) they would like to learn. The first one wants to learn SQL or O_2 but not Datalog ($K_1 = \{\neg d, s \vee o\}$). The second one wants to learn only *Datalog* or O_2but not both $K_2 = \{\neg s, d \vee o, \neg o \vee \neg d\}$. The third one wants to learn all three

[4] We denote by $<_{lex}$ the lexicographic order.

$K_3 = \{s, d, o\}$. Let $E = \{K_1 \sqcup K_2 \sqcup K_3\}$ be the corresponding belief profile. In this case, the result of the fusion will be :

- $\mathcal{F}_{Card}\mathcal{R}(E) = \{\{\neg s, d\}, \{s, d\}\}$ and $\Delta^{Card}(E) = \{\{\neg d, s \vee o, d \vee o, \neg o \vee \neg d, s, o\}, \{\neg d, s \vee o, \neg s, d \vee o, \neg o \vee \neg d, o\}\}$
- $\mathcal{F}_{\Sigma}\mathcal{R}(E) = \mathcal{F}_{Card}\mathcal{R}(E)$ and $\Delta^{\Sigma}(E) = \Delta^{Card}(E)$
- $\mathcal{F}_{Max}\mathcal{R}(E) = \{\{\neg s, d\}\}$ and $\Delta^{Max}(E) = \{\{\neg d, s \vee o, d \vee o, \neg o \vee \neg d, s, o\}\}$
- $\mathcal{F}_{GMax}\mathcal{R}(E) = \mathcal{F}_{Max}\mathcal{R}(E)$ and $\Delta^{GMax}(E) = \Delta^{Max}(E)$

We now present an implementation of RSF for the $Card$ and Σ strategies.

4 Encoding RSF in Answer Sets Programming

We now show how we construct a logic program, denoted by P_E, such that the preferred answer sets of P_E correspond to the removed sets of E.

We first show how to translate the Removed Set Fusion into a logic program, in the spirit of Niemelä in [20], in order to obtain a one-to-one correspondence between answer sets of P_E and potential removed sets of E. The key idea of the translation is to introduce for each clause, an atom which presence in the answer set corresponds to the presence of the clause in a potential removed set. We then define the notion of preferred answer set in order to perform RSF.

4.1 Translation into a Logic Program

Let $E = \{K_1, \ldots, K_n\}$ be an belief profile. The set of all positive literals of P_E is denoted by V^+. The set of all negative literals of P_E is denoted by V^-. The set of all atoms representing clauses is defined by $R^+ = \{r_c^i \mid c \in K_i\}$ and $CL(r_c^i)$ denotes the clause of K_i corresponding to r_c^i in P_E, namely $\forall r_c^i \in R^+,\ CL(r_c^i) = c$. To each answer set S of P_E, we associate the potential removed set $CL(R^+ \cap S)$.

1. The first step introduces rules in order to build a one-to-one correspondence between answer sets of P_E and interpretations of V^+. For each atom, $a \in V^+$ we introduce two rules : $a \leftarrow not\ a'$ and $a' \leftarrow not\ a$ where $a' \in V^-$ is the negative atom corresponding to a.
2. The second step excludes answer sets S which correspond to interpretations which are not models of $(K_1 \sqcup \ldots \sqcup K_n) \backslash C_i$ with $C_i = \{c | r_c \in S\}$. For each clause c of K_j such that $c = \neg b_o \vee \ldots \vee \neg b_n \vee b_{n+1} \vee \ldots \vee b_m$, we introduce the following rule $r_c^j \leftarrow b_o, \ldots, b_n, b'_{n+1}, \ldots, b'_m$

This translation differs from the one proposed in [26] for RSR since we only consider the positive atoms R^+ representing the clauses.

Example. The logic program P_E corresponding to the previous example is:

$$\begin{array}{ccc}
s \leftarrow not\ s' & s' \leftarrow not\ s & d \leftarrow not\ d' \\
d' \leftarrow not\ d & o \leftarrow not\ o' & o' \leftarrow not\ o \\
r^1_{\neg d} \leftarrow d & r^1_{s \vee o} \leftarrow s', o' & r^2_{\neg s} \leftarrow s \\
r^2_{d \vee o} \leftarrow d', o' & r^2_{\neg d \vee \neg o} \leftarrow d, o & r^3_{s} \leftarrow s' \\
r^3_{d} \leftarrow d' & r^3_{o} \leftarrow o' &
\end{array}$$

Let S be a set of atoms, we define I_S such that $I_S = \{a|a \in S\} \cup \{\neg a|a' \in S\}$. The following proposition establishes the correspondence between answer sets and models of $(K_1 \sqcup \ldots \sqcup K_n) \backslash CL(R^+ \cap S)$.

Proposition 1. *Let $E = \{K_1, \ldots, K_n\}$ be an belief profile. Let $S \subseteq V$ be a set of atoms. S is an answer set of P_E iff I_S is an interpretation of V^+ which satisfies $(K_1 \sqcup \ldots \sqcup K_n) \backslash CL(R^+ \cap S)$.*

In order to compute the answer sets corresponding to the removed sets we introduce the notion of preferred answer set according to a strategy P.

Definition 11. *Let P_E be a logic program and let S and S' be two set of atoms of P_E. S is a preferred answer set of P_E according to a strategy P iff*

- *S is an answer set of P_E;*
- *for every answer set S' of P_E, S' is not preferred to S according to P.*

The correspondence between preferred answer sets and removed sets is given by the following proposition for the strategies $Card$ and Σ.

Proposition 2. *Let $E = \{K_1, \ldots, K_n\}$ be an belief profile. X is a removed set of E according to the strategy P iff there exists a preferred answer set S of P_E according to P such that $CL(R^+ \cap S) = X$.*

Example. Let P_E be the logic program of the previous example. The collection of preferred answer sets of P_E according to the strategies $Card$ and Σ is : $\{S_1 = \{s, d', o, r^2_{\neg s}, r^3_d\}, S_2 = \{s', d', o, r^3_s, r^3_d\}\}$. Since $R^+ = \{r^1_{\neg d}, r^1_{s \vee o}, r^2_{\neg s}, r^2_{d \vee o}, r^2_{\neg d \vee \neg o}, r^3_s, r^3_d, r^3_o\}$, the removed sets are $CL(R^+ \cap S_1) = \{\neg s, d\}$ and $CL(R^+ \cap S_2) = \{s, d\}$.

4.2 Computing the Preferred Answer Sets : The RSF Algorithm

The RSF algorithm computes the preferred answer sets corresponding to the removed sets. This algorithm is a modification of Smodels algorithm that selects the preferred answer sets according to a chosen strategy P. It builds, step by step, a collection of candidate answer sets. At the end of the computation, this collection contains all the preferred answer sets corresponding to the removed sets.

The selection of the preferred answer sets is achieved thanks to the function $Condition_P(A)$, where A is a set of atoms. This function compares the current answer set candidate to the preferred answer sets previously computed. The three possible behaviors of the function $Condition_P(A)$ are:

1. A cannot lead to any preferred answer set. In this case, the computation is stopped and the algorithm backtracks;
2. A is complete and is equally preferred to the previously computed best answer sets. In this case, A is added to the collection of candidates answer sets;

3. A is complete and is preferred to the previously computed best answer sets. In this case, the collection reduced to A replaces the collection of candidates answer sets.

Another adaptation of Smodels concerns the initial heuristic of Smodels. If an atom a is selected then the atom a' cannot be deduced anymore. The only atoms that can be deduced are atoms that represent the rules r_c^i. The use of the standard heuristic leads to maximize the number of deduced r_c^i which contradicts the objective of RSF. It doesn't allow us to take advantage of the pruning of the search tree. We modify the initial heuristic in order to select the atoms that minimize the number of deduced atoms. Therefore, the first computed answer sets have greater chances to be preferred according to the chosen strategy. The new function is called $mheuristic(A)$.

Algorithm 2. Algorithm $rsf(A)$

```
A ← expand(A)
if conflict(A) then
  return false
end if
if (1) Condition_P(A) = 1 then
  return false
else if A is a subset of an already computed model then
  return false
else if A covers atom(E) then
  if (2) Condition_P(A) = 0 then
    add A to the set of solutions
    return true
  else
    (3){A} becomes the set of solutions
    return true
  end if
end if
x ← mheuristic(A)
rsf(A ∪ {x})
rsf(A ∪ {not x})
```

The adaptations of the original Smodels algorithm consist in: (i) avoiding all the subsets of R^+ leading to answer sets which removes more clauses than the removed sets; (ii) not computing several times the same subsets of literals of R^+; (iii) taking advantage of possible cuts in the search space.

5 Preliminary Experimental Study

We now present the results of a preliminary experimental study of the RSF approach. The tests were conducted on a Centrino cadenced at 1.73GHz and equipped with 1GB of RAM.

As far as we know, there is no other implementation of the fusion of propositional belief bases nor benchmarks for fusion. The following preliminary tests are not exhaustive enough to conclude about the efficiency of RSF. Nevertheless, they show the practicability of the approach. In order to be able to conclude on the efficiency of RSF, we plan to develop a more complete set of tests.

Table 1. Results for $nb = 3$, $sc = 3$ and $d = 20\%$

nc	nv	$Succes(\%)$	$Time(s)$
100	1000	100	$2,1$
200	2000	100	$7,2$
400	4000	100	$37,6$

nc	nv	$Succes(\%)$	$Time(s)$
600	6000	100	$105,2$
800	8000	100	$221,4$
1200	12000	0	$-$

Table 2. Results for $nb = 3$, $sc = 3$ and $d = 20\%$ according to nv/nc

nc	nv	$Succes(\%)$	$Time(s)$
400	200	40	$68,7$
400	400	20	$13,5$
400	800	70	$17,5$

nc	nv	$Succes(\%)$	$Time(s)$
200	100	90	$2,2$
200	200	90	$11,1$
200	400	90	$2,1$

Benchmarks are randomly generated according to several parameters: the number of bases (nb), the number of clauses in the bases (nc), the number of variables in the bases (nv), the size of clauses in the bases (sc) and a parameter that measures how belief bases differ (d).

Test bases are constructed as follows. We randomly construct an interpretation and then we randomly generate clauses that satisfy it. From one base to another, we change this interpretation according to the parameter(d) which represents the percentage of changed variables. For each set of parameters, we launched 10 different sets of the test bases. A test is considered successfull if it computes all the removed sets in less than 300 seconds and we keep the average run time of the successfull tests. The experimentation gives the percentage of successfull tests and the run time, in seconds, for the computation of all removed sets.

Table 1 shows the result of the behaviour of RSF algorithm for 3 bases consisting on ternary clauses. The RSF approach performs the fusion of 3 bases with a reasonable run time until a total number of about 3000 clauses and 8000 variables. Making nv/nc vary, table 2 shows a peak of difficulty when nv/nc approaches 1.

Analysing the running time we have observed that the heuristic for choosing the atoms is time consuming and has to be improved.

6 Conclusion

This paper presents a new approach for performing syntactic fusion of propositional beliefs bases and shows that the classical fusion operations $Card$, Σ, Max, $GMax$, initially defined at the semantic level, can be expressed within this syntactic framework.

The paper shows that RSF can be successfully encoded into answer set programming for the strategies $Card$ and Σ and proposes an implementation stemming from Smodels system. It presents a preliminary experimental study that seems promising for performing belief bases fusion on real scale applications. A future work will conduct the implementation of Max and $GMax$ strategies.

A more extensive experimentation has to be conducted on real scale applications in order to provide a more accurate evaluation of the performance of RSF. This will be conducted in a future work in the framework of an european project in the context of fusion of spatial information. Moreover, the development of a benchmarking platform for fusion will be useful, not only for testing RSF, but more globally for anyone willing to work on practical implementations of fusion operations.

Removed Set Fusion (RSF) makes it possible to efficiently implement the classical *Card* and Σ fusion operators, moreover it generalizes Removed Set Revision (RSR) since belief bases revision can be considered as the prioritized merging of two belief bases [27] and RSR amounts to the fusion of two sources according to the *Card* strategy.

Our framework could be extended according to several directions. A first extension for dealing with constraints that the merged belief base $\Delta(E)$ has to satisfy. A second extension to prioritized belief bases fusion.

A future work will detail the semantic characterization of RSF. This characterization is provided from the set of clauses of $K_1 \sqcup \cdots \sqcup K_n$ falsified by an interpretation.

Acknowledgements

Work partially supported by the European Community under project VENUS (Contract IST-034924) of the "Information Society Technologies (IST) program of the 6th FP for RTD". 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 therein.

References

1. Baral, C., Kraus, S., Minker, J., Subrahmanian, V.S.: Combining knowledge bases consisting of first order theories. In: ISMIS, pp. 92–101 (1991)
2. Revesz, P.Z.: On the semantics of theory change: arbitration between old and new information. In: 12th ACM symp. on Principles of Databases, pp. 71–92 (1993)
3. Lin, J.: Integration of weighted knowledge bases. AI 83, 363–378 (1996)
4. Revesz, P.Z.: On the semantics of arbitration. J. of Alg. and Comp. 7(2), 133–160 (1997)
5. Cholvy, L.: Reasoning about merging information. Handbook of Defeasible Reasoning and Uncertainly Management Systems 3, 233–263 (1998)
6. Konieczny, S., Pérez, R.P.: On the logic of merging. In: Proc. of KR'98, pp. 488–498 (1998)
7. Lafage, C., Lang, J.: Logical representation of preferences for group decision making. In: Proc. of KR'00, Breckenridge, pp. 457–468 (2000)
8. Konieczny, S.: On the difference between merging knowledge bases and combining them. In: Proc. of KR2000, Breckenridge, pp. 135–144. Morgan Kaufmann, San Francisco (2000)

9. Delgrande, J., Dubois, D., Lang, J.: Iterated revision as prioritized merging. In: Proc. of KR06, Lake District (UK), pp. 210–220. AAAI Press, Stanford (2006)
10. Fagin, R., Kuper, G.M., Ullman, J.D., Vardi, M.Y.: Updating logical databases. Advances in Comp. Research, 1–18 (1986)
11. Meyer, T., Ghose, A., Chopra, S.: Syntactic representations of semantic merging operations. In: IJCAI'01 Workshop on Incons. in Data and Knowledge (2001)
12. Konieczny, S., Lang, J., Marquis, P.: Distance-based merging: A general framework and some complexity results. In: Proc. of KR'02, pp. 97–108 (2002)
13. Dubois, D., Lang, J., Prade, H.: Possibilistic Logic. Handbook of Logic in Artificial Intelligence and Logic Programming 3, 439–513 (1994)
14. Benferhat, S., Dubois, D., Kaci, S., Prade, H.: Possibilistic Merging and Distance-based Fusion of Propositional Information. AMAI'02 34(1-3), 217–252 (2002)
15. Papini, O.: A complete revision function in propositionnal calculus. In: Proc. of ECAI92, pp. 339–343. J. Wiley and Sons, Chichester (1992)
16. Würbel, E., Jeansoulin, R., Papini, O.: Revision: An application in the framework of gis. In: Proc. of KR'00, Breckenridge, pp. 505–516 (2000)
17. Eiter, T., Leone, N., Mateis, C., Pfeifer, G., Scarcello, F.: The kr system dlv: progress report, comparison and benchmarks. In: Proc. of KR'98, pp. 406–417 (1998)
18. Cholewinski, P., Marek, V., Mikitiuk, A., Truszczynski, M.: Computing with default logic. AI 112, 105–146 (1999)
19. Rao, P., Sagonas, K., Swift, W.D.S., Friere, J.: Xsb: A system for efficiently computing well-founded semantics. In: Fuhrbach, U., Dix, J., Nerode, A. (eds.) LPNMR 1997. Lecture Notes in Computer Science(LNAI), vol. 1265, pp. 430–440. Springer, Heidelberg (1997)
20. Niemelä, I., Simons, P.: An implementation of stable model and well-founded semantics for normal logic programs. In: Fuhrbach, U., Dix, J., Nerode, A. (eds.) LPNMR 1997. LNCS, vol. 1265, pp. 420–429. Springer, Heidelberg (1997)
21. Linke, T.: More on nomore. In: Proc. of NMR'02 (2002)
22. Lin, J., Mendelzon, A.O.: Merging databases under constraints. IJCIS'98 7(1), 55–76 (1998)
23. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. In: Proc. of the Fifth Int. Conf. on Logic Prog., pp. 1070–1080. MIT Press, Cambridge (1988)
24. Simons, P.: Extending and implementing the stable model semantics (2000)
25. Konieczny, S.: La logique du changement - Revision et fusion de connaissances. PhD thesis, Univ. de Lille (1999)
26. Bennaim, J., Benferhat, S., Papini, O., Würbel, E.: An answer set programming encoding of prioritized removed sets revision: application to gis. In: Alferes, J.J., Leite, J.A. (eds.) JELIA 2004. LNCS (LNAI), vol. 3229, pp. 604–616. Springer, Heidelberg (2004)
27. Delgrande, J., Dubois, D., Lang, J.: Iterated revision as prioritized merging. In: Proc. of KR'06, Windermere, pp. 210–220 (2006)