Removed Sets Fusion: Performing Off The Shelf

Hué Julien [1] and **Würbel Eric** [1] and **Papini Odile** [2]

Abstract. Merging multiple sources of information is a rising subject in artificial intelligence. Most of the proposals are model-based approaches with very high computational complexity, moreover few experimentations are available. This paper proposes a framework for performing Removed Sets Fusion (RSF) of belief bases consisting of propositional formulae. It then describes the implementation of RSF which stems from Answer Set Programming (ASP) and can be performed with any ASP solver supporting the minimize statement. It finally presents an experimental study and a comparison.

1 Introduction

During the last years, the availability of distributed sources of information has significantly increased. Intelligently exploiting them remains an hard task because they often lack of structure, they can conflict and they can be redundant. The aim of fusion is to obtain a global point of view, exploiting the complementarity between sources, solving 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 [3, 26, 20, 27, 7]. 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 [17] and various operators have been defined according to whether explicit or implicit priorities are available [17], [18], [15], [8], [28]. More recent semantic approaches have been proposed, stemming from the Hamming distance [16], or based on concepts provided by mathematical morphology, like dilation and erosion [6]. The computational complexity of semantic approaches is generally very high and few implementations have been proposed. More recently, Gorogiannis and Hunter [13] proposed an implementation stemming from Binary Decision Diagrams (BDDs) and conducted an experimental study on model-based merging operators reformulated in terms of dilation.

Syntactic approaches have been proposed in propositional logic [22, 31] as well as in possibilistic logic [9, 4] which is a real advantage from the computational point of view. More recently, a syntactic approach, called Removed Sets Fusion (RSF), has been proposed for merging belief bases consisting of clauses in [14], extending the framework proposed for revision in [5]. Because it experimentally gave better results than other approaches (SAT, BDD, ...), this approach has been formalized in terms of Answer Set Programming (ASP)and the smodels system has been adapted for providing an implementation.

This paper proposes a generalization of RSF for performing syntactic fusion of belief bases consisting of propositional formulae. The paper 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 then shows that an efficient implementation of these operations, can be performed independently from the ASP solver. The main contributions of this paper are the following:

- A generalization of the Removed Sets Fusion (RSF) to the fusion of belief bases consisting of any kind of well-formed finite propositional formulae. 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 [24, 30], is extended to the subsets of formulae to remove to restore consistency. It then shows that classical fusion operations are captured within this framework since each fusion strategy is encoded by a preference relation between subsets of formulae.
- An implementation of RSF with ASP. The fusion problem is tranlated into a logic program with answer set semantics and a one-to-one correspondence between removed sets and preferred answer sets is shown. The computation of answer sets is performed with any ASP solver supporting the minimize statement.
- An experimental study which illustrates the behavior of RSF for the Σ and Max strategies, and a comparison with the results of the experimentation provided in [13].

The rest of this paper is organized as follows. Section 2 fixes the notations and gives a refresher on fusion, ASP and ASP solvers. Section 3 presents the generalization of RSF which deals with any well-formed propositional formulae. The implementation of RSF with ASP is described in Section 4. It first presents the translation into a logic program, it then shows the one-to-one correspondence between removed sets and preferred answer sets. Section 5 shows how RSF is performed off the shelf. The results of an experimental study and a comparison with the experimental results provided by [13] are presented in Section 6 before concluding in Section 7.

2 Background

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 $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ and Cn denotes the logical consequence. A *belief base* φ is a finite set of propositional formulae over $\mathcal{L}$.

[1] LSIS UMR CNRS 6168 - Équipe INCA Université de Toulon et du Var Avenue de l Université - BP20132 83957 LA GARDE CEDEX - FRANCE hue, wurbel@univ-tln.fr

[2] LSIS UMR CNRS 6168 - Équipe INCA Université de la Méditerranée - ESIL Avenue de Luminy 13288 MARSEILLE CEDEX - FRANCE odile.papini@esil.univmed.fr

2.1 Belief Merging

Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a multi-set of n consistent belief bases to be merged, Ψ is called a *belief profile*. The n belief bases $\varphi_1, \ldots, \varphi_n$ are not necessarily different and the union of belief bases, taking repetitions into account, is denoted by $\sqcup$ and their conjunction (resp. disjunction) are denoted by $\bigwedge$ (resp. $\bigvee$). For the sake of simplicity, we denote by φ the belief profile consisting of the singleton $\Psi = \{\varphi\}$.

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

There are two straighforward ways for defining $\Delta(\Psi)$ depending if the sources are conflicting or not, the classical conjunctive merging: $\Delta(\Psi) = \bigwedge_{\varphi_i \in \Psi} \varphi_i$ suitable when the sources are not conflicting and the classical disjunctive merging: $\Delta(\Psi) = \bigvee_{\varphi_i \in \Psi} \varphi_i$ appropriate in case of conflicting sources. These two opposite cases are not satisfactory, so 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 *Sum operator*, denoted by Σ, [21, 26] which follows the point of view of the majority of the belief bases of Ψ. The *Cardinality operator*, denoted by $Card$, [3] which is similar to Σ but without taking repetitions into account. The *Max-based* operator, denoted by Max [27], which tries to best satisfy all the belief bases of Ψ. The *Leximax-based* operator, denoted by $GMax$, [17] which is the lexicographic refinement of Max.

2.2 Answer Sets

A *Normal logic program* is a set of rules of the form $c \leftarrow a_1, \ldots, a_n, not\ b_1, \ldots, not\ b_m$ 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*. Let r be a rule, we introduce $head(r) = c$ and $body(r) = \{a_1, \cdots, a_n, b_1, \cdots, b_m\}$. Furthermore, let $body^+(r) = \{a_1, \cdots, a_n\}$ denotes the set of positive body atoms and $body^-(r) = \{b_1, \cdots, b_m\}$ the set of negative body atoms, it follows $body(r) = body^+(r) \cup body^-(r)$. Moreover, r^+ denotes the rule $head(r) \leftarrow body^+(r)$, 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 Π iff for any rule $r \in \Pi$, $head(r) \in X$ whenever $body(r) \subseteq X$. The smallest set of atoms which is closed under a basic program Π is denoted by $CN(\Pi)$.

The *reduct* or Gelfond-Lifschitz transformation [11], Π^X of a program Π relatively to a set X of atoms is defined by $\Pi^X = \{r^+ \mid r \in \Pi \text{ and } body^-(r) \cap X = \emptyset\}$.

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

2.3 ASP solvers

In the last decade, answer set programming has been considered as a convenient tool to handle non-monotonic reasoning. Moreover, several efficient systems, called ASP solvers, have been developed for computing answer sets, Smodels [23], XSB [25], DLV [10], NoMore [1], ASSAT [19], CMODELS [12], CLASP [2].

In order to extend the expressivity and the efficiency of ASP solvers, logic programs have been extended with new statements [29]:

- *domain definitions* allow for compactly encoding the possible values in a given domain, for example the declarations $\#domain\ possible(X), possible(1..n)$. ensure that every occurrence of the variable X will take a value from 1 to n.
- *domain restrictions*: can be added in some rules. For example, the rule $short(X) \leftarrow size(Y), X < Y$, the rule is only instantiated for X and Y such that $X < Y$.
- *cardinality optimization*: make possible to express that at most, respectively at least, some atoms should appear in the answer sets. For example the rule $h \leftarrow k\ \{a_1, \ldots, a_n\}\ l$ expresses that at least k atoms and at most l atoms among $\{a_1, \ldots, a_n\}$ should appear in the answer sets.
- *optimization statements*: allow for selecting among the possible answer sets, the ones that satisfy optimization statements like $minimize\{.\}$ or $maximize\{.\}$. For example, the statement $minimize\{a_1, \cdots, a_n\}$ allows for selecting the answer sets with as few of the given atoms $\{a_1, \cdots, a_n\}$ as possible.

At the present time, only smodels and CLASP implement all those statements.

3 RSF: dealing with any formula

We propose a new syntactic fusion framework, Removed Sets Fusion (RSF) for merging inconsistent belief bases consisting of well-formed formulae. The key idea of the approach is to remove subsets of well-formed formulae from the union of the belief bases, according to some strategy P, in order to restore consistency.

Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile such that $\varphi_1 \sqcup \ldots \sqcup \varphi_n$ is inconsistent. Removed Sets Fusion (RSF) provides, as a result of merging, a consistent subset of formulae of $\varphi_1 \sqcup \ldots \sqcup \varphi_n$.

Definition 1. *Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile such that $\varphi_1 \sqcup \ldots \sqcup \varphi_n$ is inconsistent, $X \subseteq \varphi_1 \sqcup \ldots \sqcup \varphi_n$ is a* potential removed set *of Ψ iff $(\varphi_1 \sqcup \ldots \sqcup \varphi_n) \backslash X$ is consistent.*

The number of potential removed sets is exponential with respect to the number of formulae. It is necessary to only select the most relevant ones, according to a strategy P. For every strategy P, a preorder $\leq_P$ over the potential removed sets is defined ($X \leq_P Y$ means that X is preferred to Y according the strategy P). The associated strict preorder is denoted by $<_P$.

Definition 2. *Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile such that $\varphi_1 \sqcup \ldots \sqcup \varphi_n$ is inconsistent, $X \subseteq \varphi_1 \sqcup \ldots \sqcup \varphi_n$ is a* removed set *of Ψ according to P iff* (i) *X is a potential removed set of Ψ;* (ii) *there is no $Y \subseteq \varphi_1 \sqcup \ldots \sqcup \varphi_n$ such that $Y <_P X$.*

We denote by $\mathcal{F}_P\mathcal{R}(\Psi)$ the collection of removed sets of Ψ according to P. The definition of Removed Sets Fusion is:

Definition 3. *Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile. The fusion operation $\Delta^P(\Psi)$ is defined by:*

$$\Delta^P(\Psi) = \bigvee_{X \in \mathcal{F}_P\mathcal{R}(\Psi)} \{Cn((\varphi_1 \sqcup \ldots \sqcup \varphi_n) \backslash X)\}$$

The usual merging operators $(Card, \Sigma, Max, GMax)$ are captured within our framework by encoding the preference relation between potential removed sets as given in the following table. For the $GMax$ strategy, let X be a potential removed set and φ_i be a belief base, we define $p^i(X) = |X \cap \varphi_i|$. Let L_X^Ψ be the sequence composed with every $(p^i(X))_{1 \leq i \leq n}$ in decreasing order. We denote by $\leq_{lex}$ the lexicographic ordering.

P	$X \leq_P Y$ iff
$Card$	$\lvert X\rvert \leq \lvert Y\rvert$
Σ	$\Sigma_{1 \leq i \leq n} \mid X \cap \varphi_i \mid \leq \Sigma_{1 \leq i \leq n} \lvert Y \cap \varphi_i\rvert$
Max	$max_{1 \leq i \leq n} \mid X \cap \varphi_i \mid \leq max_{1 \leq i \leq n} \mid Y \cap \varphi_i\rvert$
$GMax$	$L_X^\Psi \leq_{lex} L_Y^\Psi$

Example. *Let $\Psi = \{\varphi_1, \varphi_2, \varphi_3\}$ be a belief profile and $\varphi_1 = \{a \vee b, b\}$, $\varphi_2 = \{a \leftrightarrow b, \neg b\}$ and $\varphi_3 = \{\neg a \wedge \neg b, \neg a \vee \neg b\}$.*

$\mathcal{F}_\Sigma \mathcal{R}(\Psi) = \{\{a \vee b, b\}\}$ and $\Delta^\Sigma(\Psi) = Cn(a \leftrightarrow b, \neg b, \neg a \wedge \neg b, \neg a \vee \neg b)$.

$\mathcal{F}_{Max} \mathcal{R}(\Psi) = \{\{b, a \leftrightarrow b, \neg a \wedge \neg b\}\}$ and $\Delta^{Max}(\Psi) = Cn(a \vee b, \neg b, \neg a \vee \neg b)$.

4 Implementing RSF with ASP

The implementation constructs a logic program, denoted by Π_Ψ, such that, for any strategy P, the preferred answer sets of Π_Ψ according to P correspond to the removed sets of Ψ according to P. The fusion problem is first translated into a logic program in order to obtain a correspondence between answer sets and potential removed sets. The key idea of the translation is the introduction, for each formula, of an atom which presence in the answer set corresponds to the presence of the formula in a potential removed set. This atom reflects the syntax of the formula and thus requires the construction of intermediary atoms.

Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile. We denote by $s(\Psi)$, the belief profile made from Ψ where every rule appearing more than once is reduced to a singleton. Let $f, f^1, \ldots, f^n$ be formulae of φ_i. The set of all positive (resp. negative) literals of Π_Ψ is denoted by V^+ (resp. V^-). The set of all atoms representing formulae, called rule atoms, is defined by $R^+ = \{r_f^i | f \in \varphi_i\}$ and the intermediary atoms representing subformulae are denoted by ρ_{f^j} where f^j are subformulae of $f \in \varphi_i$. Moreover, $F_0(r_f^i)$ denotes the formula of φ_i corresponding to r_f^i in Π_Ψ, namely $\forall r_f^i \in R^+, F_0(r_f^i) = f$.

1. For every atom, $a \in V^+$, the first step introduces the rules: $a \leftarrow$ not a' and $a' \leftarrow$ not a. These rules build a correspondence between interpretations over V^+ and answer sets of the program Π_Ψ.
2. The second step introduces rule atoms which are necessary for constructing potential removed sets corresponding to any interpretation. The presence of the rule atom r_f^i in an answer set means that the formula f should be in the corresponding potential removed set. $\forall f \in \varphi_i$, the following rules are introduced according to the syntax of f[3]:
   - If $f \equiv a$, the corresponding rule is $r_f^i \leftarrow not\ a$;
   - If $f \equiv \neg f^1$, the corresponding rule is $r_f^i \leftarrow not\ \rho_{f^1}$;
   - If $f \equiv f^1 \vee \ldots \vee f^j$, the corresponding rule is $r_f^i \leftarrow \rho_{f^1}, \ldots, \rho_{f^j}$;
   - If $f \equiv f^1 \wedge \ldots \wedge f^j$, the corresponding rules are $r_f^i \leftarrow \rho_{f^1}$, $\ldots$, and $r_f^i \leftarrow \rho_{f^j}$.

[3] $a \rightarrow b$ can be considered as $\neg a \vee b$ and $a \leftrightarrow b$ as $(a \wedge b) \vee (\neg a \wedge \neg b)$. If a subformula only consists of an atom, i.e. $f^j \equiv a$, ρ_{f^j} is replaced by a

Example. *The logic program Π_Ψ corresponding to the previous example, where $\rho_1 = \rho_{a \wedge b}$ and $\rho_2 = \rho_{\neg a \wedge \neg b}$, is :*

$$\begin{array}{ccc}
a \leftarrow not\ a' & a' \leftarrow not\ a & b \leftarrow not\ b' \\
b' \leftarrow not\ b & c \leftarrow not\ c' & c' \leftarrow not\ c \\
r^1_{a \vee b} \leftarrow not\ a, not\ b & r^1_b \leftarrow not\ b & r^2_{a \leftrightarrow b} \leftarrow \rho_1, \rho_2 \\
\rho_1 \leftarrow not\ a & \rho_1 \leftarrow not\ b & \rho_2 \leftarrow a \\
\rho_2 \leftarrow b & r^2_{\neg b} \leftarrow b & r^3_{\neg a \wedge \neg b} \leftarrow a \\
r^3_{\neg a \wedge \neg b} \leftarrow b & r^3_{\neg a \vee \neg b} \leftarrow a, b &
\end{array}$$

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

Definition 4. *Let Π_Ψ be a logic program and let S and S' be two set of atoms of Π_Ψ. S is a preferred answer set of Π_Ψ according to a strategy P iff* (i) *S is an answer set of Π_Ψ;* (ii) *$\forall S' \subseteq \Pi_\Psi$, S' is not preferred to S according to P.*

From now on, we denote by I_S the interpretation over the atoms of $S \cap V^+$ which is defined as: $I_S = \{a \mid a \in S\} \cup \{\neg a \mid a' \in S\}$.

Proposition 1. *Let $\Psi = \{\varphi_1, \ldots, \varphi_n\}$ be a belief profile and S be a set of atoms. S is a answer set of Π_Ψ iff S corresponds to an interpretation I_S of V^+ which satisfies $(\varphi_1 \sqcup \ldots \sqcup \varphi_n) \backslash F_0(S \cap R^+)$.*

The following result gives the one-to-one correspondence between preferred answer sets of Π_Ψ and removed sets of Ψ according to the strategies $Card$, Σ and Max.

Proposition 2. *Let $X \subseteq (\varphi_1 \sqcup \ldots \sqcup \varphi_n)$. X be a removed set of $(\varphi_1 \sqcup \ldots \sqcup \varphi_n)$ according to P iff there exists a preferred answer set of Π_Ψ according to P such that $F_0(S \cap R^+) = X$. The proposition holds for the strategies $Card$, Σ and Max.*

Example. *Let Π_Ψ be the logic program of the previous example. The answer sets of Π_Ψ are $\{a', b', \rho_1, r^1_{a \vee b}, r^1_b\}$, $\{a', b, \rho_1, \rho_2, r^2_{\neg b}, r^2_{a \leftrightarrow b}, r^3_{\neg a \wedge \neg b}\}$, $\{a, b', \rho_1, \rho_2, r^1_b, r^2_{a \leftrightarrow b}, r^3_{\neg a \wedge \neg b}\}$, $\{a, b, \rho_2, r^2_{\neg b}, r^3_{\neg a \wedge \neg b}, r^3_{\neg a \vee \neg b}\}$. The preferred answer set for the Max strategy is $\{a, b', \rho_1, \rho_2, r^1_b, r^2_{a \leftrightarrow b}, r^3_{\neg a \wedge \neg b}\}$ and the preferred answer set for the Σ strategy is $\{a', b', \rho_1, r^1_{a \vee b}, r^1_b\}$. They correspond to the removed sets of Ψ according to Max and Σ respectively.*

The next section shows how to perform RSF thanks to any ASP solver.

5 Performing RSF *off the shelf*

A method for performing Removed Sets Fusion with any ASP solver which supports the *minimize* statement [29] consists of adding in the head of the logic program Π_Ψ, statements and rules that aims at counting rule atoms in every answer set according to the chosen strategy.

For the Σ strategy, $\Pi_\Psi^\Sigma = minimize\{r_f^i \mid r_f^i \in R^+\} \cup \Pi_\Psi$ and the following proposition holds:

Proposition (Σ). *The set of answer sets of Π_Ψ^Σ is the set of preferred answer sets of Π_Ψ for the strategy Σ.*

For the $Card$ strategy, $\Pi_\Psi^{Card} = minimize\{r_f^i \mid r_f^i \in R^+\} \cup P_{s(\Psi)}$ and the following proposition holds:

Proposition ($Card$). *The set of answer sets of Π_Ψ^{Card} is the set of preferred answer sets of Π_Ψ for the strategy $Card$.*

Considering the space complexity of the added rules, it is constant for the Σ and $Card$ strategies because the only added rule is the optimization one.

The Max strategy requires another use of the $minimize\{\}$ statement. Let S be an answer set. A first step computes $max_{1 \leq i \leq n}(|\{r_f^i | r_f^i \in S\}|)$ for S. The second step uses the $minimize\{\}$ statement to keep the smallest answer set in the sense of Max.

To compute how many formulae are removed from the most hit base, we need to know how many formulae are removed in every base of Ψ by S thanks to the following statements:

$$\Pi_\Psi^{size} = \left\{ \begin{array}{lcl} \delta_0 & : & \#domain\ possible(U). \\ \delta_1 & : & \#domain\ base(V). \\ \delta_2 & : & possible(1..m). \\ \delta_3 & : & base(1..n). \\ \alpha & : & size(U) \leftarrow U\{r_f^V | F_0(f) \in \varphi_V\}U. \end{array} \right\}$$

where m is the size of the biggest belief base. δ_i rules are domain definitions for the α rule. If there exists a base φ_V where the number of atoms r_f^V true in S is equal to U, then $size(U)$ is true in S. The greatest U for which $size(U)$ is true is computed thanks to Π_Ψ^{bound}:

$$\Pi_\Psi^{bound} = \left\{ \begin{array}{lcl} \delta_4 & : & \#domain\ possible(W). \\ \beta_1 & : & negmax(W) \leftarrow size(U), U > W. \\ \beta_2 & : & max(U) \leftarrow size(U), not\ negmax(U). \end{array} \right\}$$

β_1 determines every integer W for which there exists $U > W$ such that $size(U)$ is true. Then, $max(U)$ is true for U which is the greatest value for which $size(U)$ is true.

For the Max strategy, $\Pi_\Psi^{Max} = \Pi_\Psi^{size} \cup \Pi_\Psi^{bound} \cup \Pi_\Psi \cup minimize[max(1) = 1, \ldots, max(m) = m]$ and the following proposition holds:

Proposition (Max). *The set of answer sets of Π_Ψ^{Max} is the set of preferred answer sets of Π_Ψ according to Max.*

The space complexity of the optimization part in the Max operator is $\mathcal{O}(m \times n)$.

The $GMax$ operator compares potential removed sets based on the sequence of $|X \cap \varphi_i|$ ordered decreasingly. It is first necessary to know how many rules are removed in every base which is done by the rule:

$$\Pi_\Psi^{size_g} = \{\ \gamma_1 : size(V, U) \leftarrow U\ \{rem(V, 1), \ldots, rem(V, m)\}\ U.\ \}$$

where m stands for the size of the base φ_V.

The γ_1 rule records that, for the base φ_V, there is U removed formulae. Once these values are computed, they have to be ordered decreasingly.

$$\Pi_\Psi^{bound_g} = \left\{ \begin{array}{lcl} \alpha_0 & : & max(X_1, \ldots, X_n) \leftarrow size(Y_1, X_1), \ldots, \\ & & size(Y_n, X_n), X_1 >= X_2, \ldots, \\ & & X_{n-1} >= X_n, neq(Y_1, \ldots, Y_n). \\ \alpha_1 & : & max_1(X_1) \leftarrow max(X_1, \ldots, X_n) \\ & & , X_1 >= X_2, \ldots, X_{n-1} >= X_n. \\ \alpha_i & : & \ldots \\ \alpha_n & : & max_3(X_3) \leftarrow max(X_1, \ldots, X_n), \\ & & X_1 >= X_2, \ldots, X_{n-1} >= X_n. \end{array} \right\}$$

where X_1 is the greatest $size()$ value and X_n the smallest. Finally, the $minimize[]$ statement has to be constructed with the polynomial $X_n + X_{n-1} \times m + \ldots + X_1 \times m^{n-1}$ to optimize.

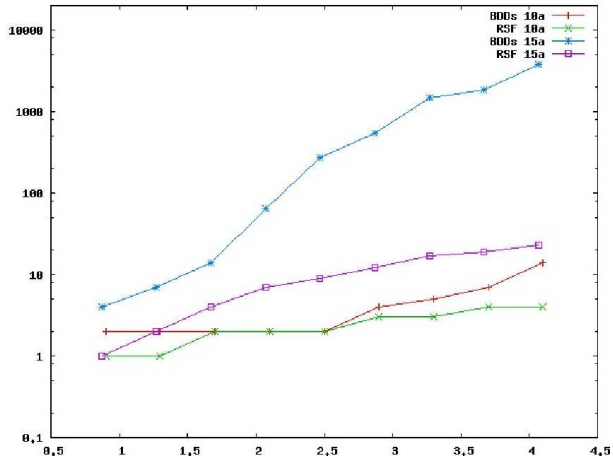
For the $GMax$ strategy, $\Pi_\Psi^{GMax} = \Pi_\Psi^{size_g} \cup \Pi_\Psi^{bound_g} \cup \Pi_\Psi \cup minimize[max_n(1) = 1, max_n(2) = 2, \ldots, max_i(1) = m^{n-i}, max_i(2) = 2 \times m^{n-i}, \ldots, max_1(n) = n \times m^{n-1}]$. and the following proposition holds:

Proposition ($GMax$). *The set of answer sets of Π_Ψ^{GMax} is the set of preferred answer sets of Π_Ψ for the strategy $GMax$.*

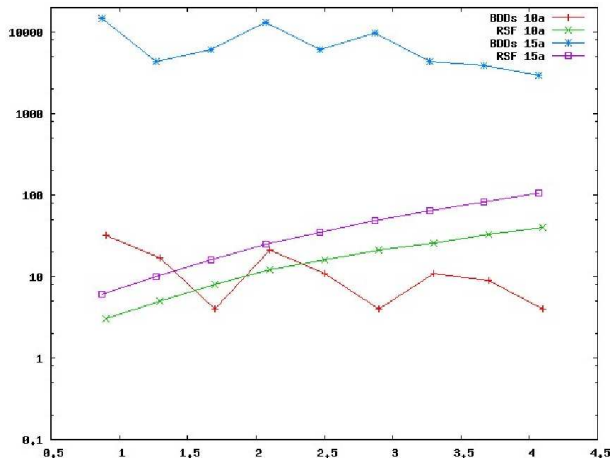
The space complexity of the optimization part in the $GMax$ operator is $\mathcal{O}(m^n \times n^n)$ which is far too high for a practical use.

6 Experimental study and comparison

We now present the results of an experimental study for the Σ and Max strategies comparing the RSF approach implemented with ASP and the model-based approach reformulated in terms of dilation implemented with BDD presented in [13]. The tests were conducted on a Centrino Duo cadenced at 1.73GHz and equipped with 2GB of RAM.



(a) The Σ strategy



(b) The Max strategy

Figure 1. Tests for two strategies

A protocol close to the one described in [13] is used. Each belief profile has seven belief bases. Belief bases are randomly generated, according to a number of atoms (na) and a number of formulae (nf). The main difference with the latter paper is

that bases consist of formulae of any kind whose depth is no greater than 3. Tests were ran for the Σ and Max strategies.

On the graphs shown in figure 1, X-axis represents the ratio $(nf)/(na)$, Y-axis represents mean running times in hundredth of seconds. Curves are drawn for 10 and 15 atoms for both strategies. On both strategies, BDDs results seem to be very unstable explaining the shape of the graphs.

What comes out from the analysis of those graphs is that RSF obtains better results when the number of atoms grows. The BDDs, on the other hand, obtains better results when the number of bases grows. Nevertheless, the fundamental difference between the two approaches forces to be very cautious with this comparison: RSF performs merging by the mean of formulae elimination, while the BDD approach described in [13] performs merging through model elimination, thus providing necessarily different results. Despite these differences, the presented comparison have the virtue to pinpoint the experimental difficulties of both approaches.

7 Conclusion

This paper generalizes the Removed Sets Fusion (RSF) approach for performing syntactic fusion of beliefs bases consisting of propositional formulae and shows that the classical fusion operations $Card$, Σ, Max, $GMax$, can be expressed within this syntactic framework.

The paper proposes an implementation of RSF stemming from ASP. RSF is translated into a logic program with answer sets semantics and the correspondence between removed sets and preferred answer sets is shown. Moreover, the paper shows how RSF can be performed with any ASP solver equipped with the *minimize* statement.

An experimentation is conducted and compared with a one stemming from BDDs, it shows that RSF gives better results.

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 the VENUS project in the context of archaeological information. A future work will detail the semantic characterization and properties of RSF.

ACKNOWLEDGEMENTS

Work partially supported by the European Community under project VENUS (Contract IST-034924) of the "Information Society Technology (IST) program of the 6th FP of RTD". The authors are solely responsible for the contents 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] C. Anger, K. Konczak, and T. Linke, 'Nomore: Nonmonotonic reasoning with logic programs', in *JELIA'02*, pp. 521–524, (2002).

[2] C. Baral, G. Brewka, and J. Schlipf, eds. *Proc. of the Ninth International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'07)*, volume 4483 of *Lecture Notes in Artificial Intelligence*, 2007.

[3] C. Baral, S. Kraus, J. Minker, and V. S. Subrahmanian, 'Combining knowledge bases consisting of first order theories.', in *ISMIS*, pp. 92–101, (1991).

[4] S. Benferhat, D. Dubois, S. Kaci, and H. Prade, 'Possibilistic Merging and Distance-based Fusion of Propositional Information', *AMAI'02*, **34**((1-3)), 217–252, (2002).

[5] J. Bennaim, S. Benferha, O. Papini, and E. Würbel, 'An answer set programming encoding of prioritized removed sets revision: application to gis', in *Proc. of JELIA'04*, pp. 604–616.

[6] I. Bloch and J. Lang. Towards mathematical morpho-logics, 2002.

[7] L. Cholvy, 'Reasoning about merging information', *Handbook of DRUMS*, **3**, 233–263, (1998).

[8] J. Delgrande, D. Dubois, and J. Lang, 'Iterated revision as prioritized merging', in *KR'06*, pp. 210–220.

[9] D. Dubois, J. Lang, and H. Prade, 'Possibilistic Logic', *in Handbook of Logic in Artificial Intelligence and Logic Programming*, **3**, 439–513, (1994).

[10] T. Eiter, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello, 'the kr system dlv: progress report, comparison and benchmarks', in *Proc. of KR'98*, pp. 406–417, (1998).

[11] M. Gelfond and V. Lifschitz, 'The stable model semantics for logic programming', in *Proc. of the 5th Int. Conf. on Log. Prog.*, pp. 1070–1080, (1988).

[12] E. Giunchiglia, Y. Lierler, and M. Maratea, 'Sat-based answer set programming.', in *Proc. of AAAI'04*, pp. 25–29, (2004).

[13] N. Gorogiannis and A. Hunter. Implementing semantic merging operators using binary decision diagrams, 2008.

[14] J. Hue, O. Papini, and Eric Würbel, 'Syntactic propositional belief bases fusion with removed sets', *Proc. of ECSQARU'07*, (2007).

[15] S. Konieczny, 'On the difference between merging knowledge bases and combining them', in *Proc of KR'00*, pp. 135–144. Morgan Kaufmann, (2000).

[16] S. Konieczny, J. Lang, and P. Marquis, 'Distance-based merging: A general framework and some complexity results', in *Proc. of KR'02*, pp. 97–108, (2002).

[17] S. Konieczny and R. Pino Pérez, 'On the logic of merging', in *Proc. of KR'98*, pp. 488–498, (1998).

[18] C. Lafage and J. Lang, 'Logical representation of preferences for group decision making', in *Proc. of KR'00*, pp. 457–468, (2000).

[19] F. Lin and Y. Zhao, 'Assat: computing answer sets of a logic program by sat solvers', *Artif. Intell.*, **157**(1-2), 115–137, (2004).

[20] J. Lin, 'Integration of weighted knowledge bases', *AI*, **83**, 363–378, (1996).

[21] J. Lin and A. O. Mendelzon, 'Merging databases under constraints', *Int. Journal of Cooperative Information Systems*, **7**(1), 55–76, (1998).

[22] T. Meyer, A. Ghose, and S. Chopra. Syntactic representations of semantic merging operations, 2001.

[23] I. Niemelä and P. Simons, 'An implementation of stable model and well-founded semantics for normal logic programs', in *Proc. of LPNMR'97*, pp. 420–429, (1997).

[24] O. Papini, 'A complete revision function in propositionnal calculus', in *Proc. of ECAI92*, ed., B. Neumann, pp. 339–343. John Wiley and Sons. Ltd, (1992).

[25] P. Rao, K. Sagonas, T.Swift, D. S. Warren, and J. Friere, 'Xsb: A system for efficiently computing well-founded semantics', in *Proc. of LPNMR'97*, pp. 430–440, (1997).

[26] P. Z. Revesz, 'On the semantics of theory change: arbitration between old and new information', 12th *ACM SIGACT-SGMIT-SIGART symposium on Principes of Databases*, 71–92, (1993).

[27] P. Z. Revesz, 'On the semantics of arbitration', *Journal of Algebra and Computation*, **7**(2), 133–160, (1997).

[28] R.Fagin, G.M.Kuper, J.D.Ullman, and M.Y.Vardi. Updating logical databases, 1986.

[29] P. Simons. Extending and implementing the stable model semantics, 2000.

[30] E. Wurbel, O. Papini, and R. Jeansoulin, 'Revision: an application in the framework of gis', in *Proc. of KR'00.*, pp. 505–516, (2000).

[31] A. Yue, W. Liu, and A. Hunter, 'Approaches to constructing a stratified merged knowledge base', in *Proc. of ECSQARU'07*, pp. 54–65, (2007).