Knowledge Base Compilation for Inconsistency Measures

Said Jabbour[1], Badran Raddaoui[2] and Lakhdar Sais[1]
[1]*CRIL - CNRS, UMR 8188, University of Artois, Lens, France*
[2]*LIAS - ENSMA EA 6315, University of Poitiers, Poitiers, France*

Keywords: Knowledge Representation, Inconsistency Measure, Conflicting Variables, Prime Implicates, Compilation.

Abstract: Measuring conflicts is recognized as an important issue for handling inconsistencies. Indeed, an inconsistency measure can be employed to support the knowledge engineer in building a consistent knowledge base or repairing an inconsistent one. Good measures are supposed to satisfy a set of rational properties. However, defining sound properties is sometimes problematic. In (Jabbour et al., 2014c), the authors proposed a new prime implicates based approach to identify the variables involved in the contradiction, and a refinement of the notion of minimal inconsistent subsets (MUSes) in propositional knowledge bases. In this article, we establish a bridge between the conflicting variables in knowledge bases and the three valued semantics by compiling each formula of the base into its prime implicates. We then extend hitting sets for MUSes to hitting sets of the set of deduced MUSes (DMUSes) based on prime implicates representation. This leads to an interesting family of inconsistency metrics.

1 INTRODUCTION

Inconsistency is often encountered, especially in case of multiple-source information. Common to most formalisms that have been studied to cope with inconsistency is the idea that some pieces of information are wrong, and thus responsible for the conflict. In this view, consistency should be recovered by removing incorrect pieces of information. This may be done, for instance by analyzing and quantifying the amount of contradictions of the set of contradictory information. Measuring conflicts has gained a considerable attention in the field of Artificial Intelligence (Bertossi et al., 2005). It is of particular importance for comparing different knowledge bases by their inconsistency levels (Grant, 1978). Indeed, it was proved useful and attractive in diverse scenarios, including software specifications (Martinez et al., 2004), e-commerce protocols (Chen et al., 2004), belief merging (Qi et al., 2005), news reports (Hunter, 2006), integrity constraints (Grant and Hunter, 2006), requirements engineering (Martinez et al., 2004), databases (Martinez et al., 2007; Grant and Hunter, 2013), semantic web (Zhou et al., 2009), and network intrusion detection (McAreavey et al., 2011), etc.

A number of logic-based inconsistency measures have been studied and there are different ways to categorize them. One way is by their dependence to the language or formula: the former aims to compute the proportion of the language affected by inconsistency (Grant, 1978; Hunter, 2002; Oller, 2004; Hunter, 2006; Grant and Hunter, 2008; Ma et al., 2010; Xiao et al., 2010a; Ma et al., 2011; Xiao and Ma, 2012; Jabbour and Raddaoui, 2013). Whilst, the latter is concerned with the minimal number of formulas that cause inconsistencies, often through minimal unsatisfiable subsets (Hunter and Konieczny, 2008; Mu et al., 2011a; Mu et al., 2012; Grant and Hunter, 2013; Jabbour et al., 2014a). Some other measures are based on both (Hunter and Konieczny, 2006; Hunter and Konieczny, 2010). Different metrics can also be classified by being formula or knowledge base oriented. For example, the inconsistency measures proposed in (Hunter and Konieczny, 2006; Hunter and Konieczny, 2010) consist in quantifying the contribution of a formula to the inconsistency of the whole knowledge base containing it, while the other mentioned measures aim to quantify the inconsistency degree of the whole knowledge base. Furthermore, some established basic properties (Hunter and Konieczny, 2010) such as *consistency, monotony, free formula independence*, are proposed to evaluate the quality of inconsistency measures.

In this work, we focus on knowledge base oriented inconsistency measures, from both the language and the formula aspects. Our aim is to investigate novel language and formula-based inconsistency measures while answering the limitations of existing ones by

Jabbour, S., Raddaoui, B. and Sais, L.
Knowledge Base Compilation for Inconsistency Measures.
DOI: 10.5220/0005824305320539
In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence (ICAART 2016) - Volume 2*, pages 532-539
ISBN: 978-989-758-172-4

satisfying the desired properties.

Inspired by the scenario given in (Xiao and Ma, 2012), suppose that there are n groups of people polling on a set of policies $\{p_1,...,p_m\}$. The poll result of each group is a set of propositional formulas. For example, the set $\{p_1 \wedge \neg p_2, p_1 \vee p_3\}$ expresses that in this group there is one voter who votes p_1 but votes against p_2, and the other voter supports either p_1 or p_3. Now consider the poll results of two groups: $\gamma_1 = \{p_1 \wedge p_2, \neg p_2\}$, $\gamma_2 = \{p_1, \neg p_1 \vee p_2, \neg p_2, p_2\}$, which are both inconsistent. Then, we can use various metrics to compare γ_1 and γ_2. By ID_4 value (Hunter, 2006), γ_1 contains one unit of inconsistency, which seems reasonable because the conflict is mere on p_2 within this group, but ID_4 treats γ_2 equivalently even though there are indeed conflicts within two subgroups. In contrast, ID_{MUS} metric (Xiao and Ma, 2012) considers that both poll results have two units of inconsistency because p_1 and p_2 are all involved in at least one subgroup with conflicts. In short, ID_4 ignores some inconsistencies for γ_2, while ID_{MUS} overestimates inconsistency in γ_1, never mentioning ID_Q (Hunter, 2002) that is always equal or larger than ID_{MUS} (Xiao and Ma, 2012). To improve these language-based measures, we revisit an interesting notion, called *conflicting variables*, from which we derive an inconsistency measure ID^c_{MUS} that can distinguish γ_1 and γ_2. Compared with ID_4 and ID_Q, the MUS based measure $I_{MI} = |MUSes(K)|$ can distinguish γ_1 and γ_2. However, as argued in (Mu et al., 2011b), I_{MI} does not satisfy the dominance property.

In this paper, we show how the compilation of a given set of formulas into its prime implicates can be used to characterize the conflicting variables involved in conflicts (Jabbour et al., 2014c). We also demonstrate that based on the same principle, we can extend many syntactic based inconsistency measures. This allows us to provide the user a way to explain its knowledge since the final formula is compiled. Furthermore, our approach allows to derive computational processes of such measures.

The paper is organized as follows: Section 2 provides basic notions and describes some inconsistency measures relevant to the present work. In Section 3, we recall the notion of conflicting variables proposed in (Jabbour et al., 2014c). In Section 4, we show how to use the compilation into prime implicates to identify the set of conflicting variables of a given knowledge base. In Section 5, we demonstrate how the DMUSes notion can be characterized based on prime implicates compilation and then propose a hitting set based approach for measuring inconsistencies.

2 PRELIMINARIES

Through this paper, We are given a propositional language $\mathcal{L}$ built over a countably infinite set of propositional symbols $\mathcal{P}$ using classical logical connectives $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$. We will use letters such as p and q to denote propositional variables, Greek letters like α and β to denote propositional formulas. The symbols $\top$ and $\bot$ denote *tautology* and *contradiction*, respectively. Sometimes, a propositional formula can be in *conjunctive normal form* (CNF), i.e., a conjunction of clauses, where a clause is a disjunction of literals and a literal is either a propositional variable (p) or its negation ($\neg p$). For a set S, $|S|$ denotes its cardinality.

We define a *knowledge base* K as a finite set of consistent propositional formulas. We denote by $Var(K)$ the set of variables occurring in K. In addition, K is inconsistent if there is a formula α such that $K \vdash \alpha$ and $K \vdash \neg\alpha$, where $\vdash$ is the deduction in classical propositional logic. If K is inconsistent, the *Minimal Unsatisfiable Subset (MUS)* of K is defined as follows:

Definition 1 (MUS)**.** *Let K be a knowledge base and $M \subseteq K$. M is a* minimal unsatisfiable (inconsistent) subset (MUS) *of K iff $M \vdash \bot$ and $\forall M' \subsetneq M$, $M' \nvdash \bot$.*

The notion of minimal inconsistent subset is also defined for CNF formula as stated in the following definition.

Definition 2. *A CNF formula α is* minimally unsatisfiable (MUS) *iff $\alpha \vdash \bot$ and for any clause $C \in \alpha$, we have $\alpha \setminus \{C\} \nvdash \bot$.*

Let $MUSes(K)$ be the set of minimal inconsistent subsets of K. Obviously, an inconsistent knowledge base K can have multiple minimal inconsistent subsets. A formula α that is not involved in any minimal inconsistent set of K is called *free formula*. The set of free formulas of K is written $free(K) = \{\alpha \mid \nexists M \in MUSes(K) \text{ s.t. } \alpha \in M\}$.

A *hitting set* of a collection of sets is a set intersecting every set of this collection. Formally,

Definition 3 (Hitting Set)**.** *H is a hitting set of a set of sets Ω if for all $S \in \Omega$, $H \cap S \neq \emptyset$. A hitting set H of Ω is irreducible if there is no other hitting set H' s.t. $H' \subset H$. A hitting set H of Ω is called a minimum hitting set, denoted as $HS_{min}(\Omega)$, if H does not strictly include any other hitting set.*

That is, taking Ω as set of MUSes involved in a knowledge base K, the minimum hitting set of Ω captures the minimum set of formulas that have to be removed from K in order to resolve the inconsistency (Reiter, 1987).

Prime implicates have been proposed as the minimal elements w.r.t. $\vdash$ in the set of all the clauses implied by a formula α, formally defined as follows:

Definition 4 (Prime Implicate)**.** *A clause C is called a* prime implicate *of a formula α if it satisfies the following conditions:*

- *$\alpha \vdash C$ holds, and*
- *for every clause C', if $\alpha \vdash C'$ and $\pi' \vdash C$ hold, then $C' \equiv C$ holds.*

$PI(\alpha)$ denotes the set of prime implicates of α.

2.1 Paraconsistent Semantics

Different from classical two-valued (true, false) semantics, multi-valued semantics (3-valued, 4-valued, LPm, and Quasi Classical) are augmented with a third truth value B to stand for the contradictory information, hence able to evaluate inconsistency. Since 3-valued, 4-valued, and LPm are the same in the context of measuring inconsistency, but differ from the one based on Quasi Classical (Xiao et al., 2010b), only 4-valued and Quasi Classical semantics will be considered throughout the paper. In the sequel, we overview some of these semantic based measures.

2.1.1 Four-valued Semantics (4-semantics)

The set of truth values for 4-valued semantics (Arieli and Avron, 1998) contains four elements: *true, false, unknown (or undefined)* and *both (or overdefined, contradictory)*. We use the symbols t, f, N, B, respectively, for these truth values. The truth value N allows to express incompleteness of information, i.e. absence of any information about truth or falsity. The four truth values together with the ordering $\preceq$ defined below form a lattice $FOUR = (\{t, f, B, N\}, \preceq)$: $f \preceq N \preceq t, f \preceq B \preceq t, N \not\preceq B, B \not\preceq N$. The 4-valued semantics of $\vee, \wedge$ connectives are defined according to the upper and lower bounds of two elements based on the ordering $\preceq$, respectively, and the operator $\neg$ is defined as $\neg t = f, \neg f = t, \neg B = B$, and $\neg N = N$.

Recall, that a 4-valued interpretation I is a *4-model* of a knowledge base K, denoted $I \models_4 K$, if for each formula $\phi \in K$, $\phi^I \in \{t, B\}$.

2.1.2 Quasi Classical Semantics (Q-semantics)

For the set of propositional variables $\mathcal{A}$, let $\mathcal{A}^{\pm}$ be a set of objects defined as $\mathcal{A}^{\pm} = \{+p, -p \mid p \in \mathcal{A}\}$, where $+p$ is a positive object, and $-p$ is a negative object.

Definition 5 (Q-models (Besnard and Hunter, 1995))**.** *Suppose $p \in \mathcal{A}$, $C_1, \ldots, C_m$ are clauses and $l_1, \ldots, l_n$ are literals. For $I \subseteq \mathcal{A}^{\pm}$, the Q-satisfiability relation $\models_Q$ is defined as follows:*

$$
\begin{array}{lll}
I \models_Q p & \textit{iff} & +p \in I; \\
I \models_Q \neg p & \textit{iff} & -p \in I; \\
I \models_Q l_1 \vee \ldots \vee l_n & \textit{iff} & [I \models_Q l_1 \textit{ or } \ldots \textit{ or } I \models_Q l_n] \\
 & & \textit{and } [\textit{ for all } i, I \models_Q \neg l_i \textit{ implies} \\
 & & I \models_Q l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_n]; \\
I \models_Q \{C_1, \ldots, C_m\} & \textit{iff} & I \models_Q C_i (1 \leq i \leq m).
\end{array}
$$

Q-semantics can also be regarded as assigning one of the four truth values $\{B, t, f, N\}$ to symbols in $\mathcal{A}$ in the following way, which enables a uniform way to define inconsistency degrees discussed below.

$$
p^I = \begin{cases} t & \text{iff } +p \in I \text{ and } -p \notin I; \\ f & \text{iff } +p \notin I \text{ and } -p \in I; \\ B & \text{iff } +p \in I \text{ and } -p \in I; \\ N & \text{iff } +p \notin I \text{ and } -p \notin I. \end{cases}
$$

2.2 Inconsistency Measures for Knowledge Bases

When a knowledge base is inconsistent the classical inference relation is trivialized, since one can deduce every formula of the language from the base. In particular, for knowledge bases that use classical logic for knowledge representation, inconsistencies render the whole knowledge base useless, due to the well-known principle *ex falso quodlibet*. Moreover, normally when given two inconsistent sets of formulas, they are not trivially equivalent. They do not contain the same information and they do not contain the same conflicts. To address this problem, numerous works on analyzing and evaluating inconsistency have been proposed, and different inconsistency measures have been studied.

More formally, an inconsistency measure is a function I that maps a knowledge base K to a non-negative real number $I(K)$ that quantifies the severity or amount of the conflict in K. We go on by stating the classical inconsistency metric that employs the set of minimal inconsistent subsets in a simple manner (Hunter and Konieczny, 2010). Minimal inconsistent sets allow us to circumscribe the minimal sub-parts of the knowledge base involved in the inconsistency. Intuitively, this measure associates a greater value for knowledge base containing more minimal inconsistent sets. In other words, the more minimal inconsistent subsets in the knowledge base K the greater the inconsistency in K.

Definition 6. *Let K be a knowledge base. Then, the I_{MI} value is defined as $I_{MI}(K) = |MUSes(K)|$.*

Recently, there is an increasing interest in quantifying inconsistency in inconsistent knowledge bases through multi-valued semantics. This is because paraconsistent reasoning systems provide a natural starting point for analyzing conflicts. Indeed, paraconsistent semantics lead to different proposals for measures of inconsistency. More formally, let I be an interpretation under i-semantics ($i = 4, Q$). Then, $Conflict(K, I) = \{p \in Var(K) \mid p^I = B\}$ is called the *conflicting set* of I with respect to the knowledge base K. Intuitively, in terms of size-wise minimality, the larger the size of the conflicting set in i-models of K, the more severe the inconsistency in K.

Definition 7 (ID_4 and ID_Q measures)**.** *Let K be a knowledge base. The 4- and Q-semantics based inconsistency degrees are defined as:*

$$ID_i(K) = \min_{I \models_i K} \frac{|Conflict(K,I)|}{|Var(K)|} \quad \textit{where } i \in \{4, Q\}.$$

Example 1. *Let $K = \{p, \neg p \vee q, \neg q \vee r, \neg r, s \vee u\}$. Consider two 4-valued models I_1 and I_2 of K defined as:*

$p^{I_1} = t, q^{I_1} = B, r^{I_1} = f, s^{I_1} = t, u^{I_1} = N;$
$p^{I_2} = B, q^{I_2} = B, r^{I_2} = B, s^{I_2} = t, u^{I_2} = N.$

Then, we have $ID_4(K) = \frac{1}{5}$. However, I_1 is not a Q-model of K. In fact, I_2 is a Q-model of K with the least amount of conflicts, thus we have $ID_Q(K) = \frac{3}{5}$.

In the literature, several logical properties have been studied to characterize inconsistency measures (Hunter and Konieczny, 2010; Jabbour et al., 2014a; Besnard, 2014). In this paper, we particularly focus on the following four postulates.

Definition 8. *For all finite sets K, K', and all formulas α, β, an inconsistency measure I should satisfy the following properties:*

(1) Monotony: $I(K) \leq I(K \cup K')$,
(2) Consistency: $I(K) = 0$ iff K is consistent,
(3) Free Formula Independence: if α is a free formula in $K \cup \{\alpha\}$, then $I(K \cup \{\alpha\}) = I(K)$,
(4) Dominance: If $\alpha \vdash \beta$ and $\alpha \nvdash \bot$, then $I(K \cup \{\beta\}) \leq I(K \cup \{\alpha\})$.

Intuitively, we want I to be a function on knowledge bases that is monotonically increasing with the inconsistency in the knowledge base. If the knowledge base K is consistent, $I(K)$ shall be minimal. The free formula independence property states that the set of formulas not involved in any minimal inconsistent subset does not influence the inconsistency measure. Finally, the dominance property states that if a formula is weakened, the knowledge base becomes less inconsistent.

3 CONFLICTING VARIABLES

In (Jabbour et al., 2014c; Jabbour et al., 2014b), the authors introduced the notion of conflicting variable in order to quantify the inconsistency of a knowledge base by circumscribing the sub-parts of the knowledge that create the contradictions. Indeed, conflicting variables are defined to catch the elements of the knowledge base that are really involved in conflicts. This leads to a more fine-grained look on how inconsistencies are distributed over the pieces of information in knowledge bases.

Definition 9 (Syntax-Based Conflicting Variable)**.** *Let K be a knowledge base and $p \in Var(K)$. p is a* conflicting variable *in K if there exists $\mathcal{S} \subseteq K$, $\mathcal{S}' \subseteq K$, and two sets of formulas $\mathcal{D}$ and $\mathcal{D}'$ such that the following conditions hold:*

(1) $|\mathcal{S}| = |\mathcal{D}|$, $\forall \alpha \in \mathcal{D}$, $\exists!\ \beta \in \mathcal{S}$ s.t. $\beta \vdash \alpha$ and $PI(\alpha) \subseteq PI(\beta)$,
(2) $\mathcal{D} \nvdash \bot$ and $\mathcal{D} \cup \{p\} \vdash \bot$,
(3) $|\mathcal{S}'| = |\mathcal{D}'|$, $\forall \alpha \in \mathcal{D}'$, $\exists!\ \beta \in \mathcal{S}'$ s.t. $\beta \vdash \alpha$ and $PI(\alpha) \subseteq PI(\beta)$,
(4) $\mathcal{D}' \nvdash \bot$ and $\mathcal{D}' \cup \{\neg p\} \vdash \bot$,
(5) $\mathcal{D} \cup \mathcal{D}'$ is a MUS.

We will denote the set of conflicting variables involved in the knowledge base K by $ConfV(K)$.

Intuitively, a conflicting variable p is a variable such that both its associated literals are logically entailed by sets of consistent logical consequences of K. For example, for the knowledge base $K = \{p \wedge q, \neg p\}$, p is expected to be a conflicting variable, but not q. A particular attention should be paid on the necessary condition $PI(\alpha) \subseteq PI(\phi)$. To illustrate this, let us consider the following knowledge base $K = \{q \wedge r, q \wedge \neg r\}$. By taking $\mathcal{S} = \{q \wedge r, q \wedge \neg r\}$ and $\mathcal{D} = \{r \vee \neg q, \neg r\}$, we have $\mathcal{D} \cup \{q\} \vdash \bot$. Now, if we consider $\mathcal{S}' = \{q \wedge r\}$ and $\mathcal{D}' = \{q\}$, then we have $\mathcal{D}' \cup \{\neg q\} \vdash \bot$. Without considering the condition $PI(\alpha) \subseteq PI(\phi)$, q would be considered conflicting while q does not contribute at all to the contradiction in K. However, by considering the condition (1) of Definition 9 (inclusion of prime implicates), it is clear that $\mathcal{D} = \{r \vee \neg q, \neg r\}$ can not be taken into account. Consequently, q is not a conflicting variable.

The notion conflicting variables proposed in Definition 9 is based on syntax. In the sequel, we recall the semantic based version of this notion in order to capture the set of problematic variables. Let us first introduce the notion of preferred i-model.

Definition 10. *For a knowledge base K and $i \in \{4, Q\}$, the set of preferred i-models of K, written $PM_i(K)$, is defined as follows: $PM_i(K) = \{I \mid I \models_i K, \forall I' \models_i K, Conflict(K, I) \subseteq Conflict(K, I')\}$.*

Definition 11 (Semantic-Based Conflicting Variable). *Let K be a knowledge base and $p \in Var(K)$. p is called a* semantic-based conflicting variable *in K if $p \in Conflict(K, \mathcal{M})$ for some $\mathcal{M} \in PM_i(K)$. We denote $ConfV_i(K)$ the set of conflicting variables under semantics i, where $i \in \{4, Q\}$.*

The above general definition allows for a range of possible conflicting variables to be defined in various ways. Indeed, by considering different paraconsistent semantics (4-semantics and Q-semantics), we get several instantiations of semantics based conflicting variables (denoted $ConfV_4(K)$ and $ConfV_Q(K)$, respectively). Note that the condition for calling a variable problematic is based on the existence of a preferred model in which the variable is valued B. That is, $p \in ConfV_i(K)$ does not require that $p \in Conflict(K, \mathcal{M})$ for *all* $\mathcal{M} \in PM_i(K)$. It is due to the fact that the set $\cap_{\mathcal{M} \in PM_i(K)} Conflict(K, \mathcal{M})$ can be empty. For instance, $\{p \vee q, \neg p, \neg q\}$ has two preferred 4-models $p^I = B$, $q^I = f$ and $p^{I'} = f$, $q^{I'} = B$ whose conflicting sets are disjoint.

Example 2. *Let $K = \{p, \neg p, p \vee r, \neg p \vee \neg r\}$ be an inconsistent knowledge base. Then, K has only one MUS, namely $\{p, \neg p\}$, so that $ConfV(K) = \{p\}$. But $ConfV_Q(K) = \{p, r\}$, since both p and r must be assigned to the value B to be a Q-model of K. In contrast, $ConfV_4(K) = \{p\}$ because there is no preferred 4-model of K assigning B to r. Now, if we consider the knowledge base $K' = \{p \vee q, \neg p, \neg q\}$, we have $ConfV(K') = ConfV_Q(K') = ConfV_4(K') = \{p, q\}$.*

As shown by the previous example, syntax-based conflicting variables always coincide with the conflicting variables under the 4-semantics. However, it is not true in the general case, as shown by the following example.

Example 3. *Given $K = \{p, \neg p \vee q, \neg q, q\}$, the single preferred 4-model I of K is $p^I = t, q^I = B$. Then, we have $ConfV_4(K) = \{q\}$. However by Definition 9, $ConfV(K) = \{p, q\}$ holds.*

Note that in this example we also have $ConfV(K) = ConfV_Q(K) = \{p, q\}$. Indeed, $ConfV_Q$ and $ConfV$ coincide due to the fact that there are reasonable deductions of both p ($p \in K$) and $\neg p$ ($\{\neg p \vee q, \neg q\} \vdash \neg p$) from the knowledge base K. While the reduction of p is trivial, we claim that the reduction of $\neg p$ should not be neglected as under $ConfV$ and $ConfV_Q$, but it is not the case of $ConfV_4$. Intuitively, $ConfV_4$ gives a cautious estimation of the conflicting variables. However, in the following example, we can see that $ConfV_Q$ estimates more variables as conflicting than $ConfV$.

Example 4. *Consider $K = \{p, \neg p, p \vee r, \neg p \vee \neg r\}$. We have $ConfV(K) = \{p\}$, since K is in CNF and it possesses a single MUS $\{p, \neg p\}$. However, $ConfV_Q(K) = \{p, r\}$ since the deduction of $\neg r$ from $\{p, \neg p \vee \neg r\}$ and that of r from $\{\neg p, p \vee r\}$ are both considered. But, the definition of conflicting variables restricts deductions of literals merely to those formulas forming a MUS (see condition (5) of Definition 9).*

The previous examples ensure that Definition 9 and Definition 11 characterize differently the notion of conflicting variable.

For what follows, we will examine more relations between syntax and semantic based conflicting variables in the context of measuring inconsistency.

4 KNOWLEDGE BASE COMPILATION FOR SEMANTIC CONFLICTING VARIABLES

In the section, we provide a deeper analysis of the semantic based characterization of the notion of conflicting variables. In particular, in semantic based applications, it could be useful to show how multi-valued semantics can be used to capture problematic variables of an inconsistent knowledge base. As mentioned earlier, the conflicting variables using multi-valued semantics are different from the ones based on syntax (see Definition 9). Therefore, in the sequel, we push further our reasoning in order to highlight the relationship between them through the prime implicates representation.

Firstly, let us recall that problematic variables in mutli-valued semantics are captured by minimal preferred models. Here, we present an approach that involves compiling each formula of the knowledge base to its set of prime implicates, and then generating conflicting variables from the new canonical representation.

Let us start by showing that multi-valued semantics is not prime implicate tolerant. That is, given two knowledge bases K and K' where K' is composed by the prime implicates of formulas in K, then K may have a multi-valued inconsistency measure different from K'. To illustrate this, let us consider the knowledge base $K = \{p \wedge (\neg p \vee q), \neg p \wedge (p \vee \neg q)\}$. Here, K possesses a unique minimal model such that $Conflict(K, I) = \{p\}$. Now, by replacing each formula in K by the set of its prime implicates, we obtain a new equivalent knowledge base $K' = \{p \wedge q, \neg p \wedge \neg q\}$. Clearly, K' contains always a unique model where $Conflict(K, I') = \{p, q\}$. Moreover, the set of conflicting variables is the same for both K and K', i.e., $ConfV(K) = ConfV(K') = \{p, q\}$. So, this example gives an intuition on how to get the link between multi-valued semantics and conflicting variables.

Let us recall that for a knowledge base K represented as a set of clauses, the set of conflicting variables $ConfV$ corresponds exactly to the variables involved in the set of minimal inconsistent subsets of K, which is equal to the union of the conflicts of the minimal preferred models of K. However, as illustrated in the previous example, such correspondence ceases to be true in general, i.e., when K is not in CNF.

In the sequel, we show how to retrieve this relationship for non clausal knowledge bases.

The following proposition states that the set of conflicting variables remains unchanged when using the prime implicates based compilation approach.

Proposition 1. *Let $K = \{\alpha_1,\dots,\alpha_n\}$ be a set of propositional formulas. Then, we have $ConfV(K) = ConfV(\{PI(\alpha_1),\dots,PI(\alpha_n)\})$.*

Proof. Let $\mathcal{D}$ and $\mathcal{D}'$ be two sets of formulas satisfying the requirements of Definition 9. Hence, $\mathcal{D}$ and $\mathcal{D}'$ form a MUS and the prime implicates of each formula in $\mathcal{D}$ and $\mathcal{D}'$ are subsets of prime implicates of a formula in K. Consequently, by compiling each formula in $\mathcal{D}$ and $\mathcal{D}'$, we deduce that $ConfV(K) = ConfV(\{PI(\alpha_1),\dots,PI(\alpha_n)\})$. □

Using Proposition 1, the following result holds.

Proposition 2. *Let $K = \{\alpha_1,\dots,\alpha_n\}$ be a set of propositional formulas. Then, we have $ConfV(K) = ConfV(\{PI(\alpha_1) \wedge \dots \wedge PI(\alpha_n)\})$.*

Proof. As shown by Proposition 1, if $\mathcal{D}$ and $\mathcal{D}'$ are two sets satisfying the requirements of Definition 9, then each formula in $\mathcal{D}$ and $\mathcal{D}'$ can be considered as conjunction of a subset of the prime implicates of a formula in K. We assume that $\mathcal{D} \cup \mathcal{D}' = \{\beta_1,\dots,\beta_m\}$. Let us consider $\beta' = \{\beta'_1,\dots,\beta'_m\}$ such that $\beta'_i \subseteq \beta_i$ (for all $1 \le i \le m$) and $\beta' \vdash \bot$. As β' is inconsistent, then there exists two literals p and $\neg p$ implied by β'. Now, assuming that any formula β'' obtained by removing any clause (prime implicate) from β'_i (for all $1 \le i \le m$) leads to consistency, i.e., $\beta'' \not\vdash \bot$, we also deduce that β' is a MUS. □

According to the previous result, we have the following interesting proposition.

Proposition 3. *Let $K = \{\alpha_1,\dots,\alpha_n\}$ be a set of propositional formulas. Then, we have $ConfV(K) = Var(MUSes(PI(\alpha_1) \wedge \dots \wedge PI(\alpha_n)))$.*

Proposition 3 provides a characterization of conflicting variables through the compilation of each initial formula into its set of prime implicates. Such compilation based approach allows us to consider the conjunction of the resulting prime implicates formulas as a single canonical formula from which conflicting variables can be computed. Indeed, using $PI(\alpha_1) \wedge \dots \wedge PI(\alpha_n)$, we have just to enumerate the set of minimal inconsistent sets in order to obtain the set of conflicting variables or to enumerate minimal preferred models and then consider the union of the underlying conflicts.

The following theorem states the characterization of conflicting variables through the compilation of knowledge bases.

Theorem 1. *Let K be a knowledge base such that $K = \{\alpha_1,\dots,\alpha_n\}$. Then, $ConfV(K) = ConfV_4(K')$ where $K' = \{PI(\alpha_1),\dots,PI(\alpha_n)\}$.*

Proof. Direct consequence of Proposition 3. □

5 HITTING SETS BASED INCONSISTENCY METRICS

Different inconsistency metrics have been defined in the light of minimal inconsistent subsets. In this section, we explore the notion of hitting set using deduced MUSes. Indeed, the notion of deduced MUS is introduced in (Jabbour et al., 2014c; Jabbour et al., 2014b) as an *original* characterization that allows us to summarize the conflict that arise in knowledge bases. More precisely, we show how inconsistency measures based on hitting set of the minimal inconsistent sets (e.g., (Mu, 2015)) can be extended to hitting sets of deduced MUSes.

For that we need some further notation.

Definition 12 (deduced MUS (Jabbour et al., 2014c))**.** *Let K be a knowledge base and $M = \langle \mathcal{S}, \mathcal{D} \rangle$ such that $\mathcal{S} = \{\phi_1,\dots,\phi_m\} \subseteq K$ and $\mathcal{D} = \{\alpha_1,\dots,\alpha_m\}$ a set of formulas ($\mathcal{S}$ is called the support of $\mathcal{D}$). M is a* MUS modulo logical deduction *of K, denoted as $DMUS(K)$, if:*

(1) $\forall (1 \le i \le m)$, $\phi_i \vdash \alpha_i$,
(2) $\forall (1 \le i \le m)$, $PI(\alpha_i) \subseteq PI(\phi_i)$,
(3) $\{\alpha_1,\dots,\alpha_m\}$ is a MUS,
(4) $\forall \alpha \in \{\alpha_1,\dots,\alpha_m\}$ there is no α' such that:
  (a) α' is weaker that α ($\alpha \vdash \alpha'$ but $\alpha' \nvdash \alpha$),
  (b) $(\mathcal{D} \setminus \{\alpha\}) \cup \{\alpha'\}$ is a MUS.

The notion of hitting sets can be extended to the set of deduced MUSes as follows.

Definition 13. *Let K be a knowledge base and $\{\langle \mathcal{S}_1, \mathcal{D}_1 \rangle,\dots,\langle \mathcal{S}_m, \mathcal{D}_m \rangle\}$ the set of its deduced MUSes. Then, H is a* hitting set *of $DMUSes(K)$ iff H is a hitting set of $\{\mathcal{D}_1,\dots,\mathcal{D}_n\}$.*

The following example shows that the size of the minimum hitting set obtained according to DMUSes (Definition 13) can differ from the one obtained with MUSes.

Example 5. *Let us consider the knowledge base* $K = \{p \wedge (\neg p \vee q), \neg p \wedge (p \vee \neg q)\}$. *We have* $|HS_{min}(MUSes(K))| = 1$ *while* $|HS_{min}(DMUSes(K))| = 2$. *Indeed, removing a formula from K allows to make it consistent. However using the DMUSes of K, we have to remove two formulas from K as it involves two independent sets.*

Generally, and similarly to the difference between MUSes and DMUSes, the minimum hitting set of DMUSes can be more larger than the one of MUSes. To illustrate such case, let us consider the knowledge base $K = \{p \wedge (\neg p \vee q_1) \wedge \ldots \wedge (\neg p \vee q_n), \neg p \wedge (p \vee \neg q_1) \wedge \ldots \wedge (p \vee \neg q_n)\}$. Obviously, the size of the minimum hitting set of MUSes of K is equal to 1 for any value of n while for DMUSes this value is equal to $n+1$. Since compiled, K will be $\{p \wedge q_1 \wedge \ldots \wedge q_n, \neg p \wedge \neg q_1 \wedge \ldots \wedge \neg q_n\}$. These differences, question us about the representation model that will be used to express agents knowledge.

As prime implicates are elementary knowledge representing each formula, the notion of DMUSes makes distinction between the sub-parts of each formula concerned by the conflict.

Proposition 4. *Let* $K = \{\alpha_1, \ldots, \alpha_n\}$ *be a knowledge base. We have,*

$$HS_{min}(DMUSes(K)) \neq HS_{min}(DMUSes(\alpha_1 \wedge \ldots \wedge \alpha_n))$$

Proof. Let us consider the two formulas $\alpha_1 = p \wedge (\neg p \vee q)$ and $\alpha_2 = \neg p \wedge (p \vee \neg q)$. We have $|HS_{min}(DMUSes(\{\alpha_1, \alpha_2\}))| = 2$ while $|HS_{min}(DMUSes(\alpha_1 \wedge \alpha_2))| = |HS_{min}(DMUSes(p \wedge (\neg p \vee q) \wedge \neg p \wedge (p \vee \neg q))| = 1$.

□

Let us recall an interesting result proved below for conflicting variables $ConfV(\{\alpha_1, \ldots, \alpha_n\}) = Var(MUSes(\{PI(\alpha_1) \wedge \ldots \wedge PI(\alpha_n))$. In similar way we have the following proposition.

Proposition 5. *Let* $K = \{\alpha_1, \ldots, \alpha_n\}$ *be a knowledge base. H is a hitting set of* $DMUSes(\{\alpha_1, \ldots, \alpha_n\})$ *iff H is a hitting set of* $PI(\alpha_1) \wedge \ldots \wedge PI(\alpha_n)$.

The result of Proposition 5 allows to have a mean to compute the set of hitting sets of DMUSes of a K. Indeed, by compiling each formula of the knowledge base into its set of prime implicates, Proposition 5 allows us to reduce the computation of the hitting sets of DMUSes of K to the computation of the hitting sets of the underlying prime implicates CNF formula, i.e., $PI(\alpha_1) \wedge \ldots \wedge PI(\alpha_n)$. This result opens interesting research avenue for practical computation using well-known available tools such as maximum satisfiability (MaxSAT) solvers.

6 CONCLUSION

In this paper, we have proposed a new framework for characterizing inconsistency. To this end, we have presented a compilation of a knowledge base that uses the prime implicates of the formulas of that base for characterizing conflicting variables. Secondly, based on the notion of conflicting variables, we have presented a family of measures aimed at helping the modeler in evaluating inconsistency in propositional knowledge bases. Lastly, we have shown how inconsistency measures based on hitting set of the minimal inconsistent sets can be extended to hitting sets of deduced MUSes. This last result opens an interesting direction for practical computation.

This work opens several other new research perspectives. First, we plan to analyze the behavior of existing inconsistency metrics based on MUSes in the light of deduced MUS. Secondly, based on the proposed inconsistency metrics, we will also work on developing a direct stepwise resolution of conflict so that inconsistency can be decreased whilst minimizing information loss. Finally, a comparative empirical evaluation of our inconsistency measures is clearly an interesting perspective.

REFERENCES

Arieli, O. and Avron, A. (1998). The value of the four values. *Artificial Intelligence*, 102:97–141.

Bertossi, L. E., Hunter, A., and Schaub, T. (2005). Introduction to inconsistency tolerance. In *Inconsistency Tolerance*, pages 1–14.

Besnard, P. (2014). Revisiting postulates for inconsistency measures. In *Logics in Artificial Intelligence - 14th European Conference, JELIA*, pages 383–396.

Besnard, P. and Hunter, A. (1995). Quasi-classical logic: Non-trivializable classical reasoning from incosistent information. In *Proc. of ECSQARU'95*, pages 44–51.

Chen, Q., Zhang, C., and Zhang, S. (2004). A verification model for electronic transaction protocols. In *APWeb*, pages 824–833.

Grant, J. (1978). Classifications for inconsistent theories. *Notre Dame Journal of Formal Logic*, 19(3):435–444.

Grant, J. and Hunter, A. (2006). Measuring inconsistency in knowledgebases. *J. Intell. Inf. Syst.*, 27(2):159–184.

Grant, J. and Hunter, A. (2008). Analysing inconsistent first-order knowledgebases. *Artif. Intell.*, 172(8-9):1064–1093.

Grant, J. and Hunter, A. (2013). Distance-based measures of inconsistency. In *ECSQARU*, pages 230–241.

Hunter, A. (2002). Measuring inconsistency in knowledge via quasi-classical models. In *AAAI/IAAI*, pages 68–73.

Hunter, A. (2006). How to act on inconsistent news: Ignore, resolve, or reject. *Data Knowl. Eng.*, 57(3):221–239.

Hunter, A. and Konieczny, S. (2006). Shapley inconsistency values. In *KR*, pages 249–259.

Hunter, A. and Konieczny, S. (2008). Measuring inconsistency through minimal inconsistent sets. In *KR*, pages 358–366.

Hunter, A. and Konieczny, S. (2010). On the measure of conflicts: Shapley inconsistency values. *Artif. Intell.*, 174(14):1007–1026.

Jabbour, S., Ma, Y., and Raddaoui, B. (2014a). Inconsistency measurement thanks to mus decomposition. In *International conference on Autonomous Agents and Multi-Agent Systems (AAMAS'14)*, pages 877–884.

Jabbour, S., Ma, Y., Raddaoui, B., and Sais, L. (2014b). On the characterization of inconsistency: A prime implicates based framework. In *26th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2014, Limassol, Cyprus, November 10-12, 2014*, pages 146–153.

Jabbour, S., Ma, Y., Raddaoui, B., and Sais, L. (2014c). Prime implicates based inconsistency characterization. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, pages 1037–1038.

Jabbour, S. and Raddaoui, B. (2013). Measuring inconsistency through minimal proofs. In *ECSQARU*, pages 290–301.

Ma, Y., Qi, G., and Hitzler, P. (2011). Computing inconsistency measure based on paraconsistent semantics. *J. Log. Comput.*, 21(6):1257–1281.

Ma, Y., Qi, G., Xiao, G., Hitzler, P., and Lin, Z. (2010). Computational complexity and anytime algorithm for inconsistency measurement. *Int. J. Software and Informatics*, 4(1):3–21.

Martinez, A. B. B., Arias, J. J. P., and and, A. F. V. (2004). On measuring levels of inconsistency in multi-perspective requirements specifications. In *PRISE'04*, pages 21–30.

Martinez, M. V., Pugliese, A., Simari, G. I., Subrahmanian, V. S., and Prade, H. (2007). How dirty is your relational database? an axiomatic approach. In *ECSQARU*, pages 103–114.

McAreavey, K., Liu, W., Miller, P., and Mu, K. (2011). Measuring inconsistency in a network intrusion detection rule set based on snort. *Int. J. Semantic Computing*, 5(3).

Mu, K. (2015). Responsibility for inconsistency. *International Journal of Approximate Reasoning*, 61:43–60.

Mu, K., Liu, W., and Jin, Z. (2011a). A general framework for measuring inconsistency through minimal inconsistent sets. *Knowl. Inf. Syst.*, 27(1):85–114.

Mu, K., Liu, W., and Jin, Z. (2012). Measuring the blame of each formula for inconsistent prioritized knowledge bases. *J. Log. Comput.*, 22(3):481–516.

Mu, K., Liu, W., Jin, Z., and Bell, D. A. (2011b). A syntax-based approach to measuring the degree of inconsistency for belief bases. *Int. J. Approx. Reasoning*, 52(7):978–999.

Oller, C. A. (2004). Measuring coherence using lp-models. *J. Applied Logic*, 2(4):451–455.

Qi, G., Liu, W., and Bell, D. A. (2005). Measuring conflict and agreement between two prioritized belief bases. In *IJCAI*, pages 552–557.

Reiter, R. (1987). A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95.

Xiao, G., Lin, Z., Ma, Y., and Qi, G. (2010a). Computing inconsistency measurements under multi-valued semantics by partial max-sat solvers. In *KR*.

Xiao, G., Lin, Z., Ma, Y., and Qi, G. (2010b). Computing inconsistency measurements under multi-valued semantics by partial max-SAT solvers. In *Proc. of KR'10*, pages 340–349.

Xiao, G. and Ma, Y. (2012). Inconsistency measurement based on variables in minimal unsatisfiable subsets. In *ECAI*, pages 864–869.

Zhou, L., Huang, H., Qi, G., Ma, Y., Huang, Z., and Qu, Y. (2009). Measuring inconsistency in dl-lite ontologies. In *Web Intelligence*, pages 349–356.