

Towards a Unified Framework for Syntactic Inconsistency Measures

Glauber De Bona

Escola Politécnica
Universidade de São Paulo
São Paulo, Brasil

John Grant

Dept. of Computer Science
and UMIACS
University of Maryland
College Park, MD, USA

Anthony Hunter

Dept. of Computer Science
University College London
London, UK

Sébastien Konieczny

CRIL
CNRS - Université d'Artois
Lens, France

Abstract

A number of proposals have been made to define inconsistency measures. Each has its rationale. But to date, it is not clear how to delineate the space of options for measures, nor is it clear how we can classify measures systematically. In this paper, we introduce a general framework for comparing syntactic inconsistency measures. It uses the construction of an inconsistency graph for each knowledgebase. We then introduce abstractions of the inconsistency graph and use the hierarchy of the abstractions to classify a range of inconsistency measures.

Introduction

Inconsistency is a key issue for operating in the real world. If we are to build computing systems that are inconsistency tolerant (i.e. systems that can handle inconsistency in information, opinions, requirements, desires, plans, etc.) then we need technologies for assessing and acting on inconsistency (Gabbay and Hunter 1991; Bertossi, Hunter, and Schaub 2004; Calvanese et al. 2008; Eiter et al. 2014).

A key aspect of inconsistency tolerance is the need to measure inconsistency so that we can better assess the nature of the inconsistency. By knowing more about the kind and degree of inconsistency, we are better able to take an appropriate action to deal with it. Application areas being developed that harness measures of inconsistency included software engineering (Mu, Liu, and Jin 2012; Borgida, Jureta, and Zamansky 2015), network intrusion detection (McA-reavey et al. 2011), reasoning with spatial and temporal information (Condotta, Raddaoui, and Salhi 2016), answer set programming (Ulbricht, Thimm, and Brewka 2016), and robotics (Costa and Martins 2016).

Numerous proposals for inconsistency measures have been made (Grant 1978; Knight 2001; Hunter 2002; Konieczny, Lang, and Marquis 2003; Hunter and Konieczny 2004; Grant and Hunter 2006; Hunter and Konieczny 2006; Ma et al. 2007; Qi and Hunter 2007; Grant and Hunter 2008; Zhou et al. 2009) and some inter-relationships established (e.g. (Grant and Hunter 2011; Thimm 2016b)). Furthermore, some axioms have been proposed for the minimal properties of such measures (Hunter and Konieczny 2006; 2010), and

some alternatives have been proposed (e.g. (Besnard 2014)) which offer some groupings of approaches. Nevertheless, our understanding of inconsistency measures lacks a general framework in which to position and compare them. To address this shortcoming, in this paper we propose a general framework for syntactic inconsistency measures.

There is currently no formal definition of syntactic inconsistency measures, though informally in the literature, they are regarded as measures defined in terms of the proof theory, rather than for instance the semantics, of a logic. The majority of syntactic inconsistency measures in the literature are based on minimal inconsistent subsets (MISs), with each such measure focusing on a different aspect of these sets. In this paper, we introduce the notion of an *inconsistency graph* to capture substantial information about the minimal inconsistent subsets of a knowledgebase. We show how and when functions on inconsistency graphs yield inconsistency measures, and identify new measures based on these functions. In the second part of the paper, we present a way of classifying inconsistency measures in a hierarchy starting with abstractions of the inconsistency graph. We then consider how existing and new inconsistency measures can be classified by this hierarchy. This is, as far we know, the first proposal for a general definition of a (large) family of inconsistency measures. Interestingly, we show how some examples of inconsistency measures, that do not appear at first to be syntactic, fit into this hierarchy.

Preliminaries

We assume a propositional language \mathcal{L} of formulas composed from a countable set of propositional variables (atoms) \mathcal{P} and the logical connectives \wedge, \vee, \neg . We use ϕ and ψ for arbitrary formulas and a, b, c, \dots for atoms. A knowledgebase K is a finite set of formulas. We let \vdash denote the classical consequence relation, and write $K \vdash \perp$ to denote that K is inconsistent. We write $\mathbb{R}^{\geq 0}$ for the set of nonnegative real numbers, $\mathbb{R}_{\infty}^{\geq 0}$ for $\mathbb{R}^{\geq 0} \cup \{\infty\}$, \mathcal{K} for the set of all knowledgebases (in some presumed language \mathcal{L}), and 2^X for the set of all subsets (the power set) of any set X .

For a knowledgebase K , $MI(K)$ is the set of minimal inconsistent subsets (MISs) of K , and $MC(K)$ is the set of maximal consistent subsets of K . Also, if $MI(K) = \{M_1, \dots, M_n\}$ then $\text{Prob}(K) = M_1 \cup \dots \cup M_n$ is the set of

problematic formulas in K , and $\text{Free}(K) = K \setminus \text{Prob}(K)$ is the set of **free** formulas in K . The set of formulas in K that are individually inconsistent is given by the function $\text{Selfcontradictions}(K) = \{\phi \in K \mid \{\phi\} \vdash \perp\}$. $\text{Atoms}(K)$ gives the atoms used as subformulas in the formulas in a knowledgebase K . In the next section we will use these functions in definitions for syntactic measures of inconsistency.

Some semantics use Priest's three valued logic (3VL) (Priest 1979) where the classical two valued semantics is augmented by a third truth value denoting inconsistency. An **interpretation** i is a function that assigns to each atom that appears in K one of three truth values: $i : \text{Atoms}(K) \rightarrow \{F, B, T\}$, where $\text{Atoms}(K)$ denotes the atoms occurring in K . The truth value of a conjunction $\phi \wedge \psi$ (respectively disjunction $\phi \vee \psi$) is the minimum (respectively maximum) of the truth value of ϕ and ψ where the truth values are ranked as $T > B > F$. The truth value of a negation $\neg\phi$ is F (respectively B, T) when ϕ is T (respectively B, F). An interpretation i whose range is $\{F, T\}$ is said to be **classical**. For an interpretation i , $\text{Conflictbase}(i) = \{a \mid i(a) = B\}$ denotes the set of atoms that are assigned the non-classical truth value B , and $\text{Truebase}(i, K) = \{\phi \in K \mid i(\phi) = T\}$ denotes the set of formulas evaluated as T in K . For a knowledgebase K we define the models as the set of interpretations where no formula in K is assigned the truth value F : $\text{Models}(K) = \{i \mid \text{for all } \phi \in K, i(\phi) = T \text{ or } i(\phi) = B\}$. Then, as a measure of inconsistency for K we define $\text{Contension}(K) = \text{Min}\{|\text{Conflictbase}(i)| \mid i \in \text{Models}(K)\}$. So the contension gives the minimal number of atoms that need to be assigned B in order to get a 3VL model of K .

When a set of classical interpretations H is such that, for each $\phi \in K$, there is an $i \in H$ such that $i(\phi) = T$, H is said to be a **hitting set** of K ¹.

Some measures are based on the **probabilistic satisfiability** problem (PSAT). Let \mathbb{I}_C denote the set of classical interpretations. A **probability mass** over a given set X is a function $\pi : X \rightarrow \mathbb{R}^{\geq 0}$ such that $\sum_{x \in X} \pi(x) = 1$. For each probability mass $\pi : \mathbb{I}_C \rightarrow \mathbb{R}^{\geq 0}$, let $P_\pi : \mathcal{L} \rightarrow \mathbb{R}$ be the function defined for all $\phi \in \mathcal{L}$ as $P_\pi(\phi) = \sum\{\pi(i) \mid i \in \mathbb{I}_C, i(\phi) = T\}$. A PSAT instance $\Gamma = \{P(\phi_i) \geq p_i \mid 1 \leq i \leq m\}$ is **satisfiable** if there is a probability mass $\pi : \mathbb{I}_C \rightarrow \mathbb{R}^{\geq 0}$ such that $P_\pi(\phi_i) \geq p_i$ for all $1 \leq i \leq m$.

Review of measures of inconsistency

An inconsistency measure assigns a nonnegative real value to every knowledgebase.

Definition 1. A function $I : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ is an **inconsistency measure** if the following two conditions hold $\forall K, K' \in \mathcal{K}$:

1. $I(K) = 0$ iff K is consistent.
2. If $K \subseteq K'$, then $I(K) \leq I(K')$.

The first requirement ensures that all and only consistent knowledgebases get measure 0. The second requirement

¹This concept of hitting sets differs from Reiter's (1987).

enforces that the addition of a set of formulas cannot decrease the inconsistency measure². The above requirements are from (Hunter and Konieczny 2006) where (1) is called *consistency* and (2) is called *monotony*. Other properties, although not satisfied by all inconsistency measures, serve to better characterise their behaviour. For instance, let K, K' denote knowledgebases and $\phi, \psi \in \mathcal{L}$:

- Independence** If $\phi \in \text{Free}(K)$, then $I(K) = I(K \setminus \{\phi\})$.
- MI-separability** If $\text{MI}(K \cup K') = \text{MI}(K) \cup \text{MI}(K')$ and $\text{MI}(K) \cap \text{MI}(K') = \emptyset$, then $I(K \cup K') = I(K) + I(K')$.
- Penalty** If $\phi \in \text{Prob}(K)$, then $I(K) > I(K \setminus \{\phi\})$.
- Super-additivity** If $K \cap K' = \emptyset$, then $I(K \cup K') \geq I(K) + I(K')$.
- Attenuation** If K, K' are MISs and $|K| < |K'|$, then $I(K) > I(K')$.
- Equal Conflict** If K, K' are MISs and $|K| = |K'|$, then $I(K) = I(K')$.
- Almost Consistency** If K_1, K_2, \dots is a sequence of MISs with $\lim_{i \rightarrow \infty} |K_i| = \infty$, then $\lim_{i \rightarrow \infty} I(K_i) = 0$.

The Independence and MI-separability properties were proposed in (Hunter and Konieczny 2008). Penalty and Super-additivity come from (Thimm 2009). Attenuation, Equal Conflict, and Almost Consistency were proposed in (Mu, Liu, and Jin 2011).

Next we recall ten inconsistency measures from the literature: the rationale for each is given below.

Definition 2. For a knowledgebase K , the inconsistency measures $I_B, I_M, I_A, I_P, I_C, I_{\#}, I_H, I_{nc}, I_{hs}$ and I_{η} are such that

- $I_B(K) = 1$ if $K \vdash \perp$ and $I_B(K) = 0$ if $K \not\vdash \perp$
- $I_M(K) = |\text{MI}(K)|$
- $I_A(K) = (|\text{MC}(K)| + |\text{Selfcontradictions}(K)|) - 1$
- $I_P(K) = |\text{Prob}(K)|$
- $I_C(K) = \text{Contension}(K)$
- $I_{\#}(K) = \begin{cases} 0 & \text{if } K \text{ is consistent} \\ \sum_{X \in \text{MI}(K)} \frac{1}{|X|} & \text{otherwise} \end{cases}$
- $I_H(K) = \min\{|X| \mid X \subseteq K \text{ and } \forall M \in \text{MI}(K) X \cap M \neq \emptyset\}$
- $I_{nc}(K) = |K| - \max\{n \mid \forall K' \subseteq K, |K'| = n \text{ implies } K' \not\vdash \perp\}$
- $I_{hs}(K) = \min\{|H| \mid H \text{ is a hitting set of } K\} - 1$ with $\min \emptyset = \infty$
- $I_{\eta}(K) = 1 - \max\{\eta \in [0, 1] \mid \{P(\phi) \geq \eta \mid \phi \in K\} \text{ is satisfiable}\}$

We explain the measures as follows: I_B (Hunter and Konieczny 2008) assigns the same value, 1, to all inconsistent knowledgebases. $I_M(K)$ (Hunter and Konieczny 2008) counts the number of minimal inconsistent subsets of K . $I_A(K)$ (Grant and Hunter 2011) counts the sum of the number of maximal consistent subsets together with the number

²We consider only absolute inconsistency measures and exclude relative inconsistency measures (ratios) such as $I_{MUS} = \frac{|\text{Atoms}(\text{MI}(K))|}{|\text{Atoms}(K)|}$ defined in (Xiao and Yue 2012).

of contradictory formulas but 1 must be subtracted to make $I(K) = 0$ when K is consistent. $I_P(K)$ (Grant and Hunter 2011) counts the number of formulas in minimal inconsistent subsets of K . $I_C(K)$ (Konieczny, Lang, and Marquis 2003; Grant and Hunter 2011) counts the minimum number of atoms that need to be assigned B amongst the 3VL models of K . $I_{\#}(K)$ (Hunter and Konieczny 2008) computes the weighted sum of the minimal inconsistent subsets of K , where the weight is the inverse of the size of the minimal inconsistent subset (and hence smaller minimal inconsistent subsets are regarded as more inconsistent than larger ones). $I_H(K)$ (Grant and Hunter 2013) is the size of the smallest set that has a non-empty intersection with every minimal inconsistent subset (originally called the d-hit inconsistency measure). $I_{nc}(K)$ (Doder et al. 2010; Thimm 2016b) finds the maximum size for a subset of K to be surely consistent and subtracts it from the size of the K . $I_{h.s.}(K)$ (the hitting set measure) (Thimm 2016b) computes the minimum number of classical interpretations $i \in \mathbb{I}_C$ needed for all formulas in K to be evaluated to T by some i ; then one is subtracted. This minimum will be infinite if there is a selfcontradiction in K . $I_{\eta}(K)$ (Knight 2001) is one minus the maximum probability lower bound one can consistently assign to all formulas in K . Each of these measures satisfies the definition of being an inconsistency measure (i.e. Definition 1) and, apart from I_{nc} , the property of independence. To see the failure of independence, consider $K = \{a \wedge \neg a, b\}$. Here b is free in K , but $I_{nc}(K) = 2$ and $I_{nc}(K \setminus \{b\}) = 1$. For other properties that hold for these measures, see (Thimm 2016a).

The use of minimal inconsistent subsets, such as for I_M , I_P , and $I_{\#}$, and the use of maximal consistent subsets such as I_A , have been proposed previously for measures of inconsistency (Hunter and Konieczny 2004; 2008). The idea of a measure that is sensitive to the number of formulas to produce an inconsistency emanates from Knight (2001) in which the more formulas needed to produce the inconsistency, the less inconsistent the set. As explored in (Hunter and Konieczny 2008), this sensitivity is obtained with $I_{\#}$. Another approach involves looking at the part of the language that is touched by the inconsistency, such as I_C , which is a semantic approach based on three-valued logic (Grant and Hunter 2011), and similar to the ones based on four-valued logic (e.g. (Hunter 2002)).

Graphical Representation of Inconsistency

We now introduce graphical representations of a knowledgebase.

Definition 3. A **bipartite graph** (or **bigraph**) is a tuple $G = \langle U, V, E \rangle$ where U and V are disjoint sets of vertices and E is the set of edges between U and V ; i.e., $e \in E$ implies $e = \{u, v\}$ for some $u \in U$ and $v \in V$.

An edge $e = \{u, v\}$ is said to **connect** (and be **incident to**) u and v , which are then called **adjacent** vertices. We write $\text{Adj}(v)$ for the set of vertices adjacent to v . A vertex is said to be **isolated** if there is no edge incident to it. Then $\text{deg}(v) = |\text{Adj}(v)|$. The **null bigraph** is $\langle \emptyset, \emptyset, \emptyset \rangle$.

We now introduce a representation for the structure of the minimal inconsistent sets of a knowledgebase.

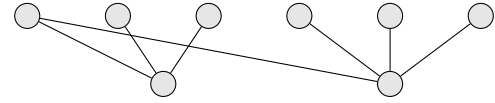
Definition 4. An **inconsistency graph** for knowledgebase K is a bigraph $\text{IG}(K) = \langle U, V, E \rangle$ such that there are bijections $b_U : U \rightarrow \text{Prob}(K)$ and $b_V : V \rightarrow \text{MI}(K)$ yielding $E = \{\{u, v\} \mid b_U(u) \in b_V(v)\}$.

We cannot tell the size of the original knowledgebase from its inconsistency graph because the free formulas are not included. To capture the free formulas, we define an augmented version, called the **augmented inconsistency graph** $\text{IG}^+(K) = \langle U, V, E \rangle$ where the only difference between the two is the inclusion of the nodes in U corresponding to free formulas, i.e. the bijection is $b_U : U \rightarrow K$. The sets V and E are the same for the inconsistency graph and the augmented inconsistency graph.

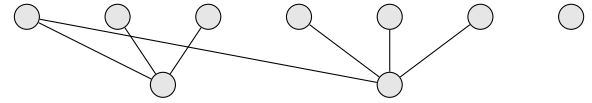
The inconsistency graph of a consistent knowledgebase K is the null bigraph. Note that, by definition, an inconsistency graph cannot have any isolated vertices.

Clearly, the inconsistency graph $\text{IG}(K)$ is a subgraph of the augmented inconsistency graph $\text{IG}^+(K)$.

Example 1. Let $K = \{a, \neg a \vee \neg b, b, \neg a \vee c, \neg c \vee d, \neg d, e \vee f\}$. Note that $\text{MI}(K) = \{\{a, \neg a \vee \neg b, b\}, \{a, \neg a \vee c, \neg c \vee d, \neg d\}\}$. The inconsistency graph for K is below.



The augmented inconsistency graph is below.



We do not put the labels on the nodes since we do not need them: only the structure of the graphs is important for our purpose.

Even though minimal inconsistent subsets of a knowledgebase are a natural starting point for considering how the inconsistency permeates a knowledgebase, we only need their structure, abstracted in the (augmented) inconsistency graphs, to capture the existing syntactic inconsistency measures.

Our first result about inconsistency graphs is a representation theorem. We show under what conditions a bigraph may represent an augmented inconsistency graph.

Theorem 1. Let $G = \langle U, V, E \rangle$ be a bigraph. Then $G = \text{IG}^+(K)$ for some knowledgebase K iff the following two conditions hold for G :

1. No vertex in V is isolated.
2. For all $v, v' \in V$, if $v \neq v'$ then $\text{Adj}(v) \not\subseteq \text{Adj}(v')$.

Corollary 1. Let $G = \langle U, V, E \rangle$ be a bigraph. Then $G = \text{IG}(K)$ for some knowledgebase K iff the following two conditions hold for G :

1. G contains no isolated vertex.
2. For all $v, v' \in V$, if $v \neq v'$ then $\text{Adj}(v) \not\subseteq \text{Adj}(v')$.

The graphs are a convenient way of conceptualizing the required abstractions that we need to consider. In addition, for small examples, the (augmented) inconsistency graphs are a simple way of visualizing the nature of the inconsistencies and of their measures. However, we are not proposing inconsistency graphs as a visualization tool for practical applications. Our aim in this paper is to show that the exploitation of this structure is the common ground of most syntactical inconsistency measures, and that it also offers a practical tool to define and classify such measures.

IG Inconsistency Measures

In this section we investigate the relation of inconsistency graphs to inconsistency measures. We show how several inconsistency measures from the literature can be represented as functions from the inconsistency graph and study when functions on inconsistency graphs yield inconsistency measures. Employing only the inconsistency graph for a knowledgebase, as the free formulas are not needed, we can calculate a number of the syntactic measures of inconsistency, without needing the knowledgebase itself, as we show next.

Proposition 1. *Let $\text{IG}(K) = \langle U, V, E \rangle$ be the inconsistency graph for a knowledgebase K . Then,*

1. $I_B(K) = \begin{cases} 0 & \text{if } V = \emptyset \\ 1 & \text{otherwise} \end{cases}$
2. $I_M(K) = |V|$
3. $I_P(K) = |U|$
4. $I_{\#}(K) = \begin{cases} 0 & \text{if } V = \emptyset \\ \sum_{v \in V} \frac{1}{|\text{deg}(v)|} & \text{otherwise} \end{cases}$
5. $I_H(K) = \min\{|X| \mid X \subseteq U \text{ and every } v \in V \text{ is adjacent to some } u \in X\}$

To characterize inconsistency measures that are functions of the inconsistency graph, such as in Proposition 1, we denote by \mathcal{G} (\mathcal{G}^+) the set of all (augmented) inconsistency graphs³

Definition 5. *An inconsistency measure $I : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ is **IG** (resp. **augmented IG**) if there is a function $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ such that $I(K) = f(\text{IG}(K))$ (resp. $I(K) = f(\text{IG}^+(K))$) for all $K \in \mathcal{K}$.*

Proposition 2. *The inconsistency measures $I_B, I_M, I_P, I_{\#}$ and I_H are IG measures.*

Since inconsistency graphs are recovered from augmented inconsistency graphs by simply discarding the isolated vertices (which correspond to the free formulas), every IG inconsistency measure is also an augmented IG measure. Even though we will show that the converse does not hold, in general most augmented IG inconsistency measures in the literature are indeed IG measures; thus we focus on the latter. This is due to the fact that IG measures are exactly the augmented IG measures satisfying the independence property, which holds for most measures in the literature (Thimm 2016a). Intuitively, the independence property guarantees that free formulas do not affect the inconsistency measure.

³Two isomorphic graphs $G_1, G_2 \in \mathcal{G}$ (resp. \mathcal{G}^+) will be considered the same element in \mathcal{G} (resp. \mathcal{G}^+).

Proposition 3. *An inconsistency measure is an IG measure iff it is an augmented IG measure satisfying independence.*

Showing that some measures are (augmented) IG measures can be fairly complicated as they require an algorithmic definition to generate them from the (augmented) inconsistency graph (in contrast to the measures considered in Proposition 1 that can be generated by simple functions).

Proposition 4. *(i) I_C is not an augmented IG measure. (ii) I_{nc} is not an IG measure, but it is an augmented IG measure. (iii) I_A, I_{hs} and I_{η} are IG measures.*

An augmented IG inconsistency measure depends only on how the formulas in a knowledgebase can be combined to form minimal inconsistent subsets and on the quantity of free formulas. Hence, inconsistency measures that are sensitive to the formulas themselves are not augmented IG measures.

The inconsistency measure I_{η} (Knight 2001), although formulated in terms of a probabilistic semantics, does not take into account the exact content of each formula in the knowledgebase, but only the consistency of its subsets.

We can use inconsistency graphs not just for classifying existing inconsistency measures but also for defining new inconsistency measures. As an illustration, we will define several new IG inconsistency measures $I(K) = f(\text{IG}(K))$, via functions f on the inconsistency graph. If $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ is a function on inconsistency graphs, we denote by $I_f : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ the function on knowledgebases defined as $I_f(K) = f(\text{IG}(K))$ for every $K \in \mathcal{K}$. Not every $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ yields a function $I_f : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ that is an inconsistency measure, for I_f must satisfy the postulates given in Definition 1.

We use the following definition of a subgraph induced by a subset of U to obtain a correspondence between subsets of a knowledgebase and subgraphs of the inconsistency graph of the knowledgebase.

Definition 6. *Let $G = \langle U, V, E \rangle$ be a bigraph and $W \subseteq U$. Let $V' = \{v \in V \mid \text{adj}(v) \subseteq W\}$. Then, let $U' = \{u \in W \mid \exists v \in V' \text{ such that } \{u, v\} \in E\}$. Finally, let $E' = \{\{u, v\} \in E \mid u \in U' \text{ and } v \in V'\}$. Then we say that $G' = \langle U', V', E' \rangle$ is U -induced by W .*

Proposition 5. *Let $G = \langle U, V, E \rangle$ be the inconsistency graph of a knowledgebase K where $K = \{\phi_1, \dots, \phi_m\}$ and each $u_i \in U$ represents the corresponding $\phi_i \in K$ and also $\text{Ml}(K) = \{\Delta_1, \dots, \Delta_n\}$ and each $v_j \in V$ represents the corresponding Δ_j . Let $K' \subseteq K$. Then $G' = \langle U', V', E' \rangle$ is the inconsistency graph of K' iff G' is the bigraph U -induced from G by W s.t. $W \subseteq U$ corresponds to the elements of K' .*

Proposition 5 gives the exact process of obtaining all the inconsistency graphs for all the subsets of a knowledgebase. This allows us to specify a necessary and sufficient condition that a function f on inconsistency graphs must satisfy in order for I_f to be an inconsistency measure.

Theorem 2. *Let $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$. $I_f : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ is an inconsistency measure iff the following two conditions hold:*

1. $f(G) = 0$ iff $G = \langle \emptyset, \emptyset, \emptyset \rangle$;

2. If $G' = \langle U', V', E' \rangle$ is U -induced by W ($W \subseteq U$) from $G = \langle U, V, E \rangle$ then $f(G') \leq f(G)$.

Corollary 2. Let $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ be such that

1. $f(G) = 0$ iff $G = \langle \emptyset, \emptyset, \emptyset \rangle$,
2. If $G' \subseteq G$ then $f(G') \leq f(G)$

Then I_f is an inconsistency measure.

Besides the inconsistency measures from Proposition 1, we can conceive of a number of IG measures I_f based on functions on inconsistency graphs.

Proposition 6. The following functions $f : \mathcal{G} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$, defined below for all $G = \langle U, V, E \rangle \in \mathcal{G}$, yield inconsistency measures $I_f : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$. (We put in parentheses the meaning for the corresponding knowledgebase.)

- $f_1(G) = |U| + |V|$ (the number of problematic formulas plus the number of minimal inconsistent subsets)
- $f_2(G) = |E|$ (the sum of the sizes of the MISs)
- $f_3(G) = |U| + |V| + |E| (f_1(G) + f_2(G))$

• $f_4(G) = \begin{cases} 0 & \text{if } U = \emptyset \\ \sum_{v \in V} \frac{\sum_{u \in \text{Adj}(v)} \text{deg}(u)}{\text{deg}(v)^2} & \text{otherwise} \end{cases}$
 (0 if K is consistent, otherwise the sum of the reciprocals of the sizes of the MISs weighted by the average number of MISs containing their elements)

• $f_5(G) = \begin{cases} 0 & \text{if } U = \emptyset \\ 1 + |\{u \in U \mid \text{deg}(u) \geq 2\}| & \text{otherwise} \end{cases}$
 (0 if K is consistent, otherwise one plus the number of formulas that are in at least two MISs)

• $f_6(G) = \begin{cases} 0 & \text{if } U = \emptyset \\ 1 + \sum_{\forall v, v' \in V, v \neq v', \text{Adj}(v) \cap \text{Adj}(v') \neq \emptyset} \frac{1}{|\text{Adj}(v) \cap \text{Adj}(v')|} & \text{otherwise} \end{cases}$
 (0 if K is consistent, otherwise one plus the sum of the reciprocals of the sizes of the intersections of each pair of minimal inconsistent subsets)

• $f_7(G) = \begin{cases} 0 & \text{if } U = \emptyset \\ \max\{\text{deg}(u) \mid u \in U\} & \text{otherwise} \end{cases}$
 (0 if K is consistent, otherwise the maximum number of minimal inconsistent subsets containing the same formula)

• $f_8(G) = \begin{cases} 0 & \text{if } U = \emptyset \\ |\{v \in V \mid \text{deg}(v) = 1\}| + \max\{\text{deg}(u) \mid u \in U\} & \text{otherwise} \end{cases}$
 (0 if K is consistent, otherwise the number of self-contradictions plus the maximum number of minimal inconsistent subsets containing the same formula)

Proposition 7. I_{f_i} satisfies independence for $1 \leq i \leq 8$, MI-separability for $i = 2$, penalty for $1 \leq i \leq 4$, super-additivity for $1 \leq i \leq 6$, attenuation for $i = 4$, equal conflict for $1 \leq i \leq 8$ and almost consistency for $i = 4$.

Consistency Graphs

The structure of the maximal consistent subsets of a knowledgebase can also be represented by bigraphs. Again, free formulas can be either included or ignored, yielding two types of bigraphs for maximal consistent subsets. To keep a

parallel with inconsistency graphs, free formulas are ignored in what is defined as the consistency graph, but allowed for in its augmented form:

Definition 7. An (augmented) consistency graph for knowledgebase K is a bigraph $\langle U, V, E \rangle$ such that there are bijections $b_U : U \rightarrow \text{Prob}(K)$ ($b_U : U \rightarrow K$) and $b_V : V \rightarrow \text{MC}(K)$ yielding $E = \{\{u, v\} \mid b_U(u) \in b_V(v)\}$.

Even though the (augmented) consistency graph of a knowledgebase is based on its maximal consistent subsets, it encodes the structure of the minimal inconsistent subsets as well. Conversely, from the (augmented) inconsistency graph one can recover the (augmented) consistency graph. Let \mathcal{G}_c (\mathcal{G}_c^+) be the set of all (augmented) consistency graphs.

Theorem 3. There is a computable bijection $h : \mathcal{G}^+ \rightarrow \mathcal{G}_c^+$ ($h : \mathcal{G} \rightarrow \mathcal{G}_c$) such that, for any $K \in \mathcal{K}$, $G = \text{IG}^+(K)$ iff $h(G) = \text{CG}^+(K)$ ($G = \text{IG}(K)$ iff $h(G) = \text{CG}(K)$).

As a consequence of Theorem 3, an inconsistency measure that is a function of the (augmented) consistency graph is also a function of the (augmented) inconsistency graph, and vice-versa. Nonetheless, some measures can be described more economically and intuitively through the consistency graph. For instance, if $\text{CG}(K) = \langle U, V, E \rangle$, $I_A(K) = |V| + |\{u \in U \mid \text{deg}(u) = 0\}| - 1$.

Abstracting the Inconsistency Graph

So far we have considered four representations for a knowledgebase: the inconsistency and consistency graphs and their augmented versions. We also investigated inconsistency measures that can be calculated from such graphs. But some inconsistency measures that can be computed from the inconsistency graph (the ones we called IG measures) do not actually use all the information provided by the inconsistency graph. This suggests that there are simpler representations that still convey enough information to calculate them. In this section we will build a hierarchy of such representations and a corresponding hierarchy for inconsistency measures based on how much information is needed to compute them.

We call such a representation an abstraction as it abstracts some information from a knowledgebase in a uniform way. We write A for the set of objects for a particular abstraction and call A an abstraction space. For instance so far we have used the set of bigraphs as an abstraction space. We also need a mapping that takes each knowledgebase to an object in the abstraction space. We do not formally define what types of operations are allowed for a mapping but note that determining if a set of formulas is consistent or inconsistent is allowed.

Definition 8. An abstraction class C (or simply a class) is a pair $C = \langle A, m_C \rangle$ where $m_C : \mathcal{K} \rightarrow A$. An inconsistency measure I is in class C if there is a function $f_C : A \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ such that $I(K) = f_C(m_C(K))$ for all knowledgebases K . We then call I a C measure.

We will sometimes write a class as C with a subscript but often it will be convenient to use just the subscript of C for the class name. Thus $C_{\text{IG}} = \text{IG} = \langle \mathcal{G}, \text{IG} \rangle$, where \mathcal{G}

Name	C	A	$m_C(K)$	$h_{IG,C}(G)$	Examples
problematic-count	PC	\mathbb{N}	$ \text{Prob}(K) $	$ U $	I_P
conflict-count	CC	\mathbb{N}	$ \text{MI}(K) $	$ V $	I_M
vertex-count	VC	\mathbb{N}^2	$\langle \text{Prob}(K) , \text{MI}(K) \rangle$	$\langle U , V \rangle$	I_P, I_M
problematic-degree	PD	$2^{\mathbb{N}^2}$	$\{\langle x, y \rangle \mid x > 0 \text{ and } x = \{\phi \in \text{Prob}(K) \text{ s.t. } \text{PFreq}(K, \phi) = y\} \}$	$\{\langle x, y \rangle \mid x > 0 \text{ and } x = \{u \in U \mid \text{deg}(u) = y\} \}$	I_{f_5}, I_{f_7}
conflict-degree	CD	$2^{\mathbb{N}^2}$	$\{\langle x, y \rangle \mid x > 0 \text{ and } x = \{M \in \text{MI}(K) \text{ s.t. } M = y\} \}$	$\{\langle x, y \rangle \mid x > 0 \text{ and } x = \{v \in V \mid \text{deg}(v) = y\} \}$	$I_{\#}$
vertex-degree	VD	$2^{\mathbb{N}^2} \times 2^{\mathbb{N}^2}$	$\langle m_{PD}(K), m_{CD}(K) \rangle$	$\langle h_{IG,PD}(G), h_{IG,CD}(G) \rangle$	I_{f_4}, I_{f_8}
edge-count	EC	\mathbb{N}	$\sum_{\Delta \in \text{MI}(K)} \Delta $	$ E $	I_{f_2}

Table 1: Classes based on abstracting the inconsistency graph $\text{IG}(K) = \langle U, V, E \rangle$. For each class C we give the full name and class name, the abstraction space, the mapping m_C from K , the function $h_{IG,C}$ from the inconsistency graph, and examples of measures defined in Definition 2 and Proposition 6. Note: $\text{PFreq}(K, \phi) = |\{M \in \text{MI}(K) \mid \phi \in M\}|$.

is the set of bigraphs and $\text{IG}(K)$ is the inconsistency graph of K , is a class and the IG inconsistency measures are exactly the ones in C_{IG} . Similarly, the class $\text{IG}^+ = \langle \mathcal{G}, \text{IG}^+ \rangle$ contains all augmented IG measures. In order to be precise in the definition of f_C , we should use $\text{Range}(m_C)$ instead of A because it is possible that not all elements of A are in $\text{Range}(m_C)$. However, since $\text{Range}(m_C) \subseteq A$, there is such $f_C : A \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ iff there is a $f_C : \text{Range}(m_C) \rightarrow \mathbb{R}_{\infty}^{\geq 0}$ with the same required property. Hence, we use both definitions interchangeably.

Another example is $C_B = \langle \{0, 1\}, I_B \rangle$, the **binary class**, where I_B was given in Definition 2. I_B is a measure in this class. In this case $f_C = \iota_{\{0,1\}}$, the identity function on $\{0, 1\}$. Due to the consistency postulate for inconsistency measures, no class can have a smaller abstraction space. In general, for any inconsistency measure I we can define the class $C_I = \langle \mathbb{R}_{\infty}^{\geq 0}, I \rangle$. A trivial case occurs if an inconsistency measure I' is obtained as a function of some I , that is, $I'(K) = g(I(K))$ in which case an I' measure is automatically an I measure. We are interested in classes that encompass genuinely different inconsistency measures, though not every class yields an inconsistency measure. For example, let $C = \langle \mathbb{N}, m_C \rangle$ where $m_C(K)$ is the number of formulas in K . There is no way to get an inconsistency measure if that is the only information stored about K . We will be interested only in classes that yield inconsistency measures. We call such a class **proper** and will discuss only proper classes.

We now formulate several classes that are intuitively more abstract than IG. In all of these cases we first show how to obtain the value from the knowledgebase and in parentheses we indicate how to obtain it from the inconsistency graph: (1) PC counts the number of problematic formulas (the size of U); (2) CC counts the number of minimal inconsistent subsets (the size of V); (3) VC combines the PC and CC values into a pair of numbers; (4) PD counts for each positive integer n the number of formulas (if not 0) that are in n minimal inconsistent subsets (the number of vertices in U that have degree n) to form a set of ordered pairs of positive integers; (5) CD counts for each positive integer n the num-

ber of minimal inconsistent subsets (if not 0) that contain n formulas (the number of vertices in V that have degree n) to form a set of ordered pairs of positive integers; (6) VD combines the PD and CD sets into an ordered pair; and (7) EC counts the sum of the sizes of the minimal inconsistent sets (the size of E). Table 1 gives all the definitions along with examples of inconsistency measures in each class.

We compare classes with respect to how abstract they are. Consider the class $CC = \langle \mathbb{N}, |\text{MI}| \rangle$ which is obtained by abstracting from the knowledgebase its number of minimal inconsistent subsets. But we can also calculate the number of minimal inconsistent subsets from the inconsistency graph by counting the number of vertices in V . However, given the number of inconsistent subsets we cannot construct the inconsistency graph. Thus, intuitively, the class IG is less abstract than the class CC. The most general class is the one where we retain the entire knowledgebase. We denote this class as $U = \langle \mathcal{K}, \iota_{\mathcal{K}} \rangle$ where $\iota_{\mathcal{K}}$ is the identity function on \mathcal{K} and call it the **universal class**. Next we formally define the abstraction relation on classes.

Definition 9. A class $C = \langle A, m_C \rangle$ is **less or equally abstract** as the class $C' = \langle A', m_{C'} \rangle$, denoted $C \preceq C'$, if the relation $h_{C,C'} = \{\langle m_C(K), m_{C'}(K) \rangle \in A \times A' \mid K \in \mathcal{K}\}$ is a function. We say that C and C' are **equally abstract** written $C \sim C'$ if both $C \preceq C'$ and $C' \preceq C$ hold. Furthermore, C is said to be **less abstract** than C' , denoted $C \prec C'$, when $C \preceq C'$ but $C \not\sim C'$.

Proposition 8. The relation \preceq is reflexive and transitive.

Next we present several results that we will use to find abstraction relations between classes. We start with a characterization for equally abstract classes.

Proposition 9. $C = \langle A, m_C \rangle$ and $C' = \langle A', m_{C'} \rangle$ are equally abstract iff $h_{C,C'}$ and $h_{C',C}$ are inverse functions.

Corollary 3. If $C \preceq C'$ and $h_{C,C'}$ is not one-to-one then $C \prec C'$.

As we next show the abstraction relation between classes extends to their inconsistency measures.

Proposition 10. *If $C' \preceq C$ (C' is less or equally abstract as C) then every C inconsistency measure is also a C' inconsistency measure.*

Corollary 4. *If $C \preceq C'$ and there is a C inconsistency measure that is not a C' inconsistency measure then $C \prec C'$.*

Example 2. $U = \langle \mathcal{K}, \iota_{\mathcal{K}} \rangle \preceq C = \langle A, m_C \rangle$ (the universal class is less or equally abstract as any class) because $h_{U,C} \subseteq \mathcal{K} \times A$ is a function, defined as $h_{U,C}(\iota_{\mathcal{K}})(K) = m_C(K) = m_C(\iota_{\mathcal{K}}(K))$ for all $K \in \mathcal{K}$. We place U , the least abstract class, at the top of the hierarchy.

Going in the opposite direction, we wish to show that for any proper class C , C is less or equally abstract than the binary class (i.e. $C = \langle A, m_C \rangle \preceq B = \langle \{0,1\}, I_B \rangle$). For this purpose we need to show that $h_{C,B} \subseteq A \times \{0,1\}$ is a function. As C is a proper class, there must be an inconsistency measure, say I_C in C , and so we have $I_C(K) = f_C(m_C(K))$, for a function $f_C : A \rightarrow \mathbb{R}_{\geq 0}^{\infty}$. Hence, $h_{C,B} : \text{Range}(m_C) \rightarrow \{0,1\}$ is a function defined as $h_{C,B}(m_C(K)) = 0$ if $f_C(m_C(K)) = 0$, $h_{C,B}(m_C(K)) = 1$ otherwise.

Although we do not require the independence property for inconsistency measures, most measures I in the literature have this property so that $I(K) = I(\text{Prob}(K))$. Thus we can formulate the **problematic class** $P = \langle \mathcal{K}, \text{Prob} \rangle$ where Prob was defined in the Preliminaries section.

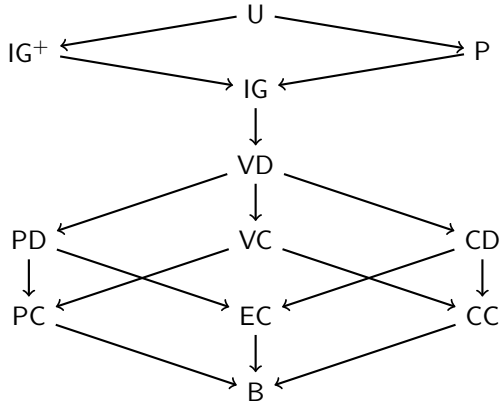


Figure 1: The abstraction hierarchy of inconsistency measures classes

The abstraction relation among the classes presented in this section (including Table 1) is given by the next result and summarized in Figure 1.

Theorem 4. *The following abstraction relations hold: (1) $U \prec IG^+, P$; (2) $IG^+, P \prec IG$; (3) $IG \prec VD$; (4) $VD \prec PD, VC, CD$; (5) $PD \prec PC, EC$; (6) $VC \prec PC, CC$; (7) $CD \prec EC, CC$; (8) $PC, EC, CC \prec B$.*

It is straightforward to show how developments of the measures reviewed in this paper (such as the vector-based approach of (Mu et al. 2011)) can fit into this hierarchy.

Let us note that there are some related works that have already used similar graphs (Jabbour, Ma, and Raddaoui

2014) and hypergraphs (Jabbour et al. 2016) for inconsistency measures, but only to define new inconsistency measures. The main point of our paper is that inconsistency graphs can be used to compare and classify most syntactical inconsistency measures.

The general framework we have proposed complements the framework proposed in (Thimm 2016b) where the expressivity of an inconsistency measure is quantified in terms of the size of its range when its domain is restricted somehow. In contrast, the abstraction classes we propose are meant to capture the information needed to compute a measure. When a class C is said to be less abstract than $C' = \langle A', m' \rangle$, this means that the information conveyed by the elements of A' is not enough to compute all measures in C . Nevertheless, it may be the case that some measures in C' have higher Thimm's expressivity (in all 4 dimensions) than measures in C that are not in C' .

Conclusion and Future Work

In this paper we propose the first, as far as we know, general definition of a large family of inconsistency measures, thanks to the notion of inconsistency graphs, that leads to a hierarchy of these measures based on how much information is needed to compute them. Even though we have intentionally avoided defining syntactic inconsistency measures in this paper, the definition of IG measures seems to formally capture this intuitive idea. Evidence for this is that most inconsistency measures called syntactic are indeed IG measures, but no measure said to be semantic is so.

The framework in this paper can be exploited in a precise way to compare different proposals for inconsistency measures, and in particular of the information that is required to calculate a given measure, and to identify new measures. So for a given application, an awareness of where a measure resides in the hierarchy will facilitate a clearer understanding of what is being measured, and whether that is appropriate for that application. Future work includes the classification of non-syntactic inconsistency measures (e.g. (Konieczny, Lang, and Marquis 2003)).

Acknowledgments. GDB is supported by CNPq grant PDE 200780/2015-8. AH is partly supported by EPSRC grant EP/N008294/1.

References

- Bertossi, L.; Hunter, A.; and Schaub, T. 2004. Introduction to inconsistency tolerance. In *Inconsistency Tolerance*, LNCS. Springer. 1–14.
- Besnard, P. 2014. Revisiting postulates for inconsistency measures. In *Proc. of European Conference on Logics in Artificial Intelligence (JELIA'14)*, 383–396.
- Borgida, A.; Jureta, I.; and Zamansky, A. 2015. Towards a general formal framework of coherence management in requirements engineering. In *Proc. of International Requirements Engineering Conference (RE'15)*, 274–277.
- Calvanese, D.; De Giacomo, G.; Lembob, D.; Lenzerinib, M.; and Rosati, R. 2008. Inconsistency tolerance in p2p

- data integration: An epistemic logic approach. *Information Systems* 33(4–5):360–384.
- Condotta, J.; Raddaoui, B.; and Salhi, Y. 2016. Quantifying conflicts for spatial and temporal information. In *Proc. of International Conference on Principles of Knowledge Representation and Reasoning (KR'16)*, 443–452.
- Costa, D., and Martins, M. 2016. Intelligent-based robot to deal with contradictions. In *Proc. of International Conference on Autonomous Robot Systems and Competitions (ICARSC'16)*, 199–204.
- Doder, D.; Rašković, M.; Marković, Z.; and Ognjanović, Z. 2010. Measures of inconsistency and defaults. *International Journal of Approximate Reasoning* 51(7):832–845.
- Eiter, T.; Fink, M.; Schlller, P.; and Weinzier, A. 2014. Finding explanations of inconsistency in multi-context systems. *Artificial Intelligence* 216:233–274.
- Gabbay, D., and Hunter, A. 1991. Making inconsistency respectable 1: A logical framework for inconsistency in reasoning. In *Foundations of Artificial Intelligence Research*, volume 535, 19–32. Springer.
- Grant, J., and Hunter, A. 2006. Measuring inconsistency in knowledgebases. *Journal of Intelligent Information Systems* 27:159–184.
- Grant, J., and Hunter, A. 2008. Analysing inconsistent first-order knowledge bases. *Artificial Intelligence* 172:1064–1093.
- Grant, J., and Hunter, A. 2011. Measuring the good and bad in inconsistent information. In *Proc. of International Joint Conference on AI (IJCAI'11)*, 2632–2637.
- Grant, J., and Hunter, A. 2013. Distance-based measures of inconsistency. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU'13)*, 230–241.
- Grant, J. 1978. Classifications for inconsistent theories. *Notre Dame Journal of Formal Logic* 19:435–444.
- Hunter, A., and Konieczny, S. 2004. Approaches to measuring inconsistent information. In *Inconsistency Tolerance*, LNCS. Springer. 189–234.
- Hunter, A., and Konieczny, S. 2006. Shapley inconsistency values. In *Proc. of 10th International Conference on Knowledge Representation (KR'06)*, 249–259.
- Hunter, A., and Konieczny, S. 2008. Measuring inconsistency through minimal inconsistent sets. In *Proc. of 11th International Conference on Knowledge Representation (KR'08)*, 358–366.
- Hunter, A., and Konieczny, S. . 2010. On the measure of conflicts: Shapley inconsistency values. *Artificial Intelligence* 174:1007–1026.
- Hunter, A. 2002. Measuring inconsistency in knowledge via quasi-classical models. In *Proc. of National Conference on Artificial Intelligence (AAAI'02)*, 68–73. MIT Press.
- Jabbour, S.; Ma, Y.; Raddaoui, B.; Sais, L.; and Salhi, Y. 2016. A mis partition based framework for measuring inconsistency. In *Proc. of International Conference on Principles of Knowledge Representation and Reasoning (KR'16)*.
- Jabbour, S.; Ma, Y.; and Raddaoui, B. 2014. Inconsistency measurement thanks to mus decomposition. In *Proc. of International Conference on Autonomous agents and multi-agent systems (AAMAS'14)*, 877–884.
- Knight, K. 2001. Measuring inconsistency. *Journal of Philosophical Logic* 31:77–98.
- Konieczny, S.; Lang, J.; and Marquis, P. 2003. Quantifying information and contradiction in propositional logic through epistemic actions. In *Proc. of International Joint Conference on Artificial Intelligence (IJCAI'03)*, 106–111.
- Ma, Y.; Qi, G.; Hitzler, P.; and Lin, Z. 2007. Measuring inconsistency for description logics based on paraconsistent semantics. In *Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU'07)*, 30–41.
- McAreavey, K.; Liu, W.; Miller, P.; and Mu, K. 2011. Measuring inconsistency through minimal inconsistent sets. *Knowledge and Information Systems* 27(1):85–114.
- Mu, K.; Liu, W.; Jin, Z.; and Bell, D. 2011. A syntax-based approach to measuring the degree of inconsistency for belief bases. *International Journal of Approximate Reasoning* 52:978–999.
- Mu, K.; Liu, W.; and Jin, Z. 2011. A general framework for measuring inconsistency through minimal inconsistent sets. *Knowledge and Information Systems* 27(1):85–114.
- Mu, K.; Liu, W.; and Jin, Z. 2012. A blame-based approach to generating proposals for handling inconsistency in software requirements. *International Journal of Knowledge and Systems Science* 3(1):1–17.
- Priest, G. 1979. The logic of paradox. *Journal of Philosophical Logic* 8(1):219–241.
- Qi, G., and Hunter, A. 2007. Measuring incoherence in description logic-based ontologies. In *Proc. of International Semantic Web Conference (ISWC'07)*, 381–394.
- Reiter, R. 1987. A theory of diagnosis from first principles. *Artificial Intelligence* 32:57–95.
- Thimm, M. 2009. Measuring inconsistency in probabilistic knowledge bases. In *Proc. of International Conference on Uncertainty in Artificial Intelligence (UAI'09)*, 530–537.
- Thimm, M. 2016a. On the compliance of rationality postulates for inconsistency measures: A more or less complete picture. *KI-Künstliche Intelligenz* 1–9.
- Thimm, M. 2016b. On the expressivity of inconsistency measures. *Artificial Intelligence* 234:120–151.
- Ulbricht, M.; Thimm, M.; and Brewka, G. 2016. Measuring inconsistency in answer set programs. In *Proc. of European Conference on Logics in Artificial Intelligence (JELIA'16)*, 577–583.
- Xiao, G., and Yue, M. 2012. Inconsistency measurement based on variables in minimal unsatisfiable subsets. In *Proc. of European Conference on Artificial Intelligence (JELIA'12)*, 864–869.
- Zhou, L.; Huang, H.; Qi, G.; Ma, Y.; Huang, Z.; and Qu, Y. 2009. Measuring inconsistency in dl-lite ontologies. In *Proc. of International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'09)*, 349–356.