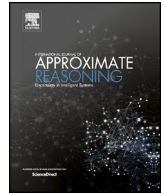




Contents lists available at ScienceDirect

## International Journal of Approximate Reasoning

journal homepage: [www.elsevier.com/locate/ijar](http://www.elsevier.com/locate/ijar)

## Semantic inconsistency measures using 3-valued logics

John Grant<sup>a</sup>, Anthony Hunter<sup>b,\*</sup><sup>a</sup> Department of Computer Science and UMIACS, University of Maryland, USA<sup>b</sup> Department of Computer Science, University College London, UK

## ARTICLE INFO

## Article history:

Received 14 December 2022

Received in revised form 16 February 2023

Accepted 20 February 2023

Available online 24 February 2023

## Keywords:

Inconsistency measures

Inconsistency analysis

Inconsistency tolerance

Semantic inconsistency measures

3-valued logic

## ABSTRACT

AI systems often need to deal with inconsistencies. One way of getting information about inconsistencies is by measuring the amount of information in the knowledgebase. In the past 20 years numerous inconsistency measures have been proposed. Many of these measures are syntactic measures, that is, they are based in some way on the minimal inconsistent subsets of the knowledgebase. Very little attention has been given to semantic inconsistency measures, that is, ones that are based on the models of the knowledgebase where the notion of a model is generalized to allow an atom to be assigned a truth value that denotes contradiction. In fact, only one nontrivial semantic inconsistency measure, the contension measure, has been in wide use. The purpose of this paper is to define a class of semantic inconsistency measures based on 3-valued logics. First, we show which 3-valued logics are useful for this purpose. Then we show that the class of semantic inconsistency measures can be developed using a graphical framework similar to the way that syntactic inconsistency measures have been studied. We give several examples of semantic inconsistency measures and show how they apply to three useful 3-valued logics. We also investigate the properties of these inconsistency measures and show their computation for several knowledgebases.

© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Inconsistency is a key issue for intelligent agents operating in the real world. Routinely, we are faced with inconsistencies when we go about our everyday lives. So if we are to build software systems that are capable of handling inconsistency in information, opinions, requirements, desires, plans, etc., for instance when making decisions, then we need to understand the formal nature of inconsistency when assessing and acting on it [19,9,22].

A key aspect of understanding inconsistency is the ability to measure it. Application areas being investigated for inconsistency measures include software engineering [56,36], network intrusion detection [38], ontology systems [47,55], knowledgebase systems [37,44,42], databases [16,4,5,43,35,34], temporal information [25], spatio-temporal information [12,23], probabilistic information [13,46], finance [18], process specifications [11], and answer set programming [39,53].

A number of proposals for inconsistency measures have been made [24,33,30,32,26,20,40,47,21,55] and some inter-relationships established (e.g. [22,50]). Some comparison of these approaches has been possible through the proposal for axioms for the minimal properties of such inconsistency measures [27,28,6]).

\* Corresponding author.

E-mail address: [anthony.hunter@ucl.ac.uk](mailto:anthony.hunter@ucl.ac.uk) (A. Hunter).

More recently, a general framework for comparing syntactic inconsistency measures has been proposed [14]. Syntactic measures are based on the inconsistent subsets of a knowledgebase and as a consequence these measures are sensitive to the syntactic form of the formulae. The framework uses the construction of an inconsistency bigraph for each knowledgebase, together with a hierarchy of the abstractions to classify a range of inconsistency measures.

However, there is a variety of measures that cannot be captured by inconsistency graphs including what are sometimes called semantic measures (e.g. [30–32,20,21]). As semantic inconsistency measures are defined using models, and inconsistent knowledgebases have no models (in classical 2-valued logic), semantic inconsistency measures use models with more than two truth values. These are models of the knowledgebase obtained using paraconsistent logics such as Priest’s 3-valued LPm logic [45], Belnap’s 4-valued logic [3], and quasi-classical logic [8]. In these logics, the notion of a model is generalized to allow an atom to be assigned a truth value that denotes contradiction. The contension measure for a knowledgebase, which is the most common kind of semantic inconsistency measure, is the lowest number of atoms assigned a contradictory truth value from amongst the models of the knowledgebase using LPm logic.

To address the lack of a general framework for semantic measures, the aim of this paper is to delineate semantic measures and to investigate the space of possible semantic measures. We propose a framework that can be used to generate and study a class of semantic inconsistency measures. We show how 3-valued logics can be used to define a class of semantic inconsistency measures, give several examples, and show their computation on several knowledgebases.

Considering semantic measures is important because the way that they analyse inconsistency differs from the more widely considered syntactic measures. As argued in [28], two key approaches for measuring inconsistency are **formula-centric** measures, which are exemplified by measures based on the minimal inconsistent subsets of the knowledgebase, and **atom-centric** measures, which are exemplified by measures based on the number of atoms that are assigned an inconsistent truth value in a 3-valued logic for paraconsistent reasoning.

A key advantage of atom-centric measures is the ability to “look inside” formulas. So all of the subformulas are taken into account when determining the amount of inconsistency of a formula or set of formulas. This is in contrast with the formula-centric measures for which the inconsistency measure of a formula does not take into account its subformulas other than to determine whether a subset of formulas is inconsistent or not. For example, knowledgebases  $\Delta_1 = \{a \wedge \neg a \wedge b\}$  and  $\Delta_2 = \{a \wedge b, \neg a \wedge b, \neg a\}$  are treated differently when considering the minimal inconsistent subsets, but are treated the same way when considering 3-valued models. This is because  $\Delta_1$  has one minimal inconsistent subset and  $\Delta_2$  has two minimal inconsistent subsets, but the semantic approach recognizes that the problem is with the atom  $a$  and not with the atom  $b$  in both knowledgebases. More generally, the semantic approach can avoid syntax sensitivity where uncontroversial atoms may be “locked” into a controversial formula (such as in  $\Delta_1$ ). The atom-centric approach therefore appears to offer some benefits for better understanding inconsistency in some applications.

Despite the utility of an atom-centric approach offered by semantic measures, our understanding of them is less developed than the formula-centric approach offered by syntactic measures. As we will review in this paper, there are few specific proposals in the literature. Also, their general nature is less well understood. This is particularly remarkable given that there is an interesting space of alternative definitions if we consider different logics that can be used as the basis of the measures. This in turn raises the question of what are the possible logics we can use, and if the choice of the logic affects the measure. We will attempt to investigate these and further questions.

The plan for the rest of this paper is as follows. In Section 2, we provide the needed background for inconsistency measures, give several examples, and discuss properties, called rationality postulates, for inconsistency measures. Section 3 introduces syntactic inconsistency bigraphs that were proposed in [14] for providing a framework for syntactic inconsistency measures. Then, in Section 4, we show which 3-valued logics are useful for defining semantic inconsistency measures. We present the concept of minimal L-atomic subsets, the key concept for defining semantic inconsistency measures in Section 5. Section 6 describes semantic inconsistency bigraphs used to define semantic inconsistency measures in a way that is similar to the use of syntactic inconsistency bigraphs for syntactic inconsistency measures. The concept of semantic inconsistency measures is defined and illustrated in Section 7. The satisfaction of postulates for semantic inconsistency measures is investigated in Section 8. Then, in Section 9, we add some new postulates that are more appropriate for semantic inconsistency measures. The paper is summarized in Section 10.

## 2. Inconsistency measures and rationality postulates

We assume a propositional language  $\mathcal{L}(\mathcal{A})$  of formulas composed from a countable set of propositional atoms  $\mathcal{A}$  and the logical connectives  $\wedge, \vee, \neg$ . We use  $\phi$  and  $\psi$  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. Also  $\text{Atoms}(K)$  gives the atoms used in the formulas of  $K$ . We write  $\mathcal{K}$  for the set of all knowledgebases (in some presumed language  $\mathcal{L}(\mathcal{A})$ ). Also we write  $\mathbb{R}^{\geq 0}$  for the set of nonnegative real numbers and  $\mathbb{R}_{\infty}^{\geq 0}$  for  $\mathbb{R}^{\geq 0} \cup \{\infty\}$ .

For a knowledgebase  $K$ ,  $\text{MI}(K)$  is the set of minimal inconsistent subsets (MISs) of  $K$ . Also, if  $\text{MI}(K) = \{M_1, \dots, M_n\}$  then  $\text{Problematic}(K) = M_1 \cup \dots \cup M_n$  is the set of **problematic formulas** in  $K$ , and  $\text{Free}(K) = K \setminus \text{Problematic}(K)$  is the set of **free formulas** in  $K$ . The set of formulas in  $K$  that are individually inconsistent is given by the function  $\text{Selfcontradictions}(K) = \{\phi \in K \mid \{\phi\} \vdash \perp\}$ .

In classical 2-valued logic an interpretation  $i$  for  $\mathcal{L}(\mathcal{A})$  maps each atom to a truth value, that is,  $i : \mathcal{A} \rightarrow \{T, F\}$ . By using the standard truth tables in Table 1 we extend  $i$  from the atoms to arbitrary formulas.

**Table 1**  
The truth tables for classical 2-valued logic.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T
F	F	F
T	F	T

$\vee$	F	T
F	F	T
T	T	T

We write  $i \models \phi$  if the truth value of  $\phi$  is  $T$  for interpretation  $i$  and say that  $i$  is a *model* of  $\phi$ . Then,  $i$  is a **model** of knowledgebase  $K$ , written  $i \models K$  iff  $i \models \phi$  for all  $\phi \in K$ . When dealing with a specific knowledgebase  $K$ , the atoms not in  $K$  are irrelevant; hence we will just deal with the portion of the interpretation that is applied to  $\text{Atoms}(K)$ . For knowledgebases  $K$  and  $K'$  we say that  $K$  (**logically**) **implies**  $K'$  iff every model of  $K$  is a model of  $K'$ . Then,  $K$  is (**logically**)-**equivalent** to  $K'$  iff  $K$  implies  $K'$  and  $K'$  implies  $K$ .

An *inconsistency measure* is a function that assigns a nonnegative real value or infinity to every knowledgebase. The only requirement is that all and only inconsistent knowledgebases have a positive value.

**Definition 1.** A function  $I : \mathcal{K} \rightarrow \mathbb{R}_{\infty}^{\geq 0}$  is an **inconsistency measure** iff the following condition holds for all  $K \in \mathcal{K}$ :

**Consistency**  $I(K) = 0$  iff  $K$  is consistent.

Next we review seven inconsistency measures that have been presented by various researchers in the literature. These are key examples of syntactic measures of inconsistency. We leave coverage of semantic measures of inconsistency until Section 7.

**Definition 2.** For a knowledgebase  $K$ , the inconsistency measures  $I_B, I_M, I_P, I_R, I_a, I_{P(r)}$  and  $I_{a(r)}$  are such that

- $I_B(K) = 1$  if  $K \vdash \perp$  and  $I_B(K) = 0$  if  $K \not\vdash \perp$
- $I_M(K) = |\text{MI}(K)|$
- $I_P(K) = |\text{Problematic}(K)|$
- $I_R(K) = \min\{|X| \mid X \subseteq K \text{ and } \forall M \in \text{MI}(K)(X \cap M \neq \emptyset)\}$
- $I_a(K) = |\text{Atoms}(\text{Problematic}(K))|$
- $I_{P(r)}(K) = \frac{I_P(K)}{|K|}$
- $I_{a(r)}(K) = \frac{I_a(K)}{|\text{Atoms}(K)|}$

We explain the measures as follows:  $I_B$  is a limiting case that does not distinguish among inconsistent knowledgebases. Thinking of an MIS as a unit of inconsistency,  $I_M$  counts the number of units of inconsistency.  $I_P$  counts the number of formulas directly involved in at least one inconsistency.  $I_R$ , the repair measure, is the minimal number of formulas whose deletion restores consistency.  $I_a$  counts the number of atoms directly involved in at least one inconsistency.  $I_{P(r)}$  finds the ratio of the number of formulas involved in at least one inconsistency with the total number of formulas.  $I_{a(r)}$  finds the ratio of the number of distinct atoms involved in at least one inconsistency with the total number of distinct atoms in the KB.

$I_B, I_M, I_P, I_R, I_{P(r)}$  can be calculated from the structure of the minimal inconsistent subsets as we will show in the next section. Such inconsistency measures are called *syntactic* inconsistency measures.  $I_{P(r)}$  and  $I_{a(r)}$  relativize  $I_P$  and  $I_a$  respectively to obtain a ratio. Such inconsistency measures are called *relative* inconsistency measures and were investigated in [7].  $I_B, I_M, I_P, I_R,$  and  $I_a$  measure the totality, in some way, of the inconsistency. Such inconsistency measures are called *absolute* inconsistency measures.

To evaluate how appropriate a function is to be considered an inconsistency measure, rationality postulates were introduced. A *rationality postulate* is a condition that intuitively an inconsistency measure should satisfy. However, some postulates may be appropriate for some types of measures, but not others. In particular, the distinction between postulates for absolute and relative inconsistency measures is considered in detail in [7]. Here we list three examples each of three types of postulates: ones that deal with deletion from a knowledgebase, ones that deal with the union of knowledgebases, and ones that use logical implication.

**Definition 3.** The following are rationality postulates for propositional inconsistency measures.

**Monotony** If  $K \subseteq K'$  then  $I(K) \leq I(K')$ .

**Free-Formula Independence** If  $\phi \in \text{Free}(K)$ , then  $I(K) = I(K \setminus \{\phi\})$ .

**Penalty** If  $\phi \in \text{Problematic}(K)$ , then  $I(K) > I(K \setminus \{\phi\})$ .

**Super-Additivity** If  $K \cap K' = \emptyset$ , then  $I(K \cup K') \geq I(K) + I(K')$ .

**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  $I(K \cup K') = I(K) + I(K')$ .

**Relative Separability** If  $I(K) < I(K')$  (resp.  $I(K) = I(K')$ ) and  $\text{Atoms}(K) \cap \text{Atoms}(K') = \emptyset$  then  $I(K) < I(K \cup K') < I(K')$  (resp.  $I(K) = I(K \cup K') = I(K')$ ).

**Table 2**  
Status of postulates for inconsistency measures.

Inconsistency measures	$I_B$	$I_M$	$I_P$	$I_R$	$I_a$	$I_{P(r)}$	$I_{a(r)}$
Monotony	✓	✓	✓	✓	✓	✗	✗
Free-Formula Independence	✓	✓	✓	✓	✓	✗	✗
Penalty	✗	✓	✓	✗	✗	✓	✗
Super-Additivity	✗	✓	✓	✓	✗	✗	✗
MI-Separability	✗	✓	✗	✗	✗	✗	✗
Relative Separability	✗	✗	✗	✗	✗	✓	✓
Dominance	✓	✗	✗	✗	✗	✗	✗
Exchange	✓	✗	✗	✗	✗	✗	✗
Adjunction Invariance	✓	✗	✗	✗	✗	✗	✗

✓: satisfied, ✗: not satisfied.

**Dominance** If  $\phi \not\vdash \perp$  and  $\phi \vdash \psi$  then  $I(K \cup \{\phi\}) \geq I(K \cup \{\psi\})$ .

**Exchange** If  $K' \not\vdash \perp$  and  $K' \equiv K''$  then  $I(K \cup K') = I(K \cup K'')$ .

**Adjunction Invariance**  $I(K \cup \{\phi, \psi\}) = I(K \cup \{\phi \wedge \psi\})$ .

All of these postulates, except for Relative Separability, are discussed in [52] where references are given for them. Relative Separability was introduced in [7] for relative measures, meaning that the proportion of inconsistency in the union of two knowledgebases is between the proportion of inconsistency in them separately. We can group these postulates three ways. Monotony, Free-Formula Independence, and Penalty consider the case where one knowledgebase is a subset of another one. Super-Additivity, MI-Separability, and Relative Separability consider the case of the union of knowledgebases. Finally, Dominance, Exchange, and Adjunction Invariance concern cases where logical implication is used. Table 2 shows the status of the postulates defined in Definition 3 for the inconsistency measures of Definition 2. Most of these results are taken from [52].

### 3. Syntactic inconsistency bigraphs

The use of bigraphs to study primarily syntactic inconsistency measures was introduced in [14]. This section reviews the basic definitions of bigraphs and their use for syntactic inconsistency measures.

**Definition 4.** Bigraphs and labeled bigraphs.

1. A **bigraph** is a tuple  $G = \langle U, V, E \rangle$  where  $U$  and  $V$  are sets,  $U \cap V = \emptyset$ , and  $E = U \times V$ .
2. A **labeled bigraph** is a tuple  $LG = \langle U, V, E, L \rangle$  where the first 3 components form a bigraph and  $L$  is a function  $L : U \cup V \rightarrow S$ , where  $S$  is a set of labels.

We will use  $u$  with subscripts for elements of  $U$  and  $v$  with subscripts for elements of  $V$ . For ease of presentation, when we draw bigraphs we will assume that  $U = \{u_1, \dots, u_m\}$  and  $V = \{v_1, \dots, v_n\}$ . We draw a bigraph using small circles without labels representing the vertices in two rows: the ones on the upper row represent the vertices in  $U$ , that we assume to be  $u_1, \dots, u_m$  from left to right, while the ones on the lower row represent the vertices in  $V$ , that we assume to be  $v_1, \dots, v_n$  from left to right, and the edges are lines between the circles. For a labeled bigraph we use boxes for the vertices with the labels inside. The ordering of the vertices is arbitrary:  $U$  and  $V$  are sets, not tuples; while a reordering would make the picture look different, it would represent the same (labeled) bigraph.

The following is a brief review of commonly used definitions about bigraphs. We write  $G$  for an arbitrary bigraph. An edge  $e = (u, v)$  is said to **connect (be incident to)**  $u$  and  $v$ , which are then called **adjacent** vertices. We write  $\text{Adj}(u)$  (resp.  $\text{Adj}(v)$ ) for the set of vertices adjacent to  $u$  (resp.  $v$ ). A vertex is said to be *isolated* if there is no edge incident to it. Then  $\text{Deg}(u) = |\text{Adj}(u)|$  (resp.  $\text{Deg}(v) = |\text{Adj}(v)|$ ).

Consider a situation where a set of elements  $S$  is given as well as a property  $P$  that a nonempty subset may satisfy. We are interested in minimal subsets of  $S$  that have property  $P$ , that is, subsets  $S_1 \subseteq S$  such that there is no  $S_2 \subset S_1$  such that  $S_2$  has property  $P$ . For such a case we construct a bigraph (the unlabeled version)  $G = \langle U, V, E \rangle$  where  $U = S$ ,  $V$  is the minimal subsets that satisfy  $P$  and  $E$  is the set of pairs  $(u, v)$  where  $v$  represents a minimal subset that contains  $u$ . We call such a bigraph a **minimal subset bigraph**. Note that in a minimal subset bigraph no vertex in  $V$  can be isolated because it cannot represent the empty set. Also, on account of minimality, for any  $v \neq v'$ ,  $\text{Adj}(v) \not\subseteq \text{Adj}(v')$ .

A minimal subset bigraph was introduced in [14] where  $S$  is the set of elements of a knowledgebase and the subset property is “inconsistent”. Thus  $U = K$  and  $V = \text{MI}(K)$ . In this paper we call such a bigraph a **syntactic inconsistency bigraph**. Next we give the definition both for the labeled and the unlabeled versions.

The following definition defines labeled syntactic inconsistency bigraphs.

**Definition 5.** The **labeled syntactic inconsistency bigraph** for KB  $K$ , where  $K = \{\phi_1, \dots, \phi_m\}$  and  $\text{MI}(K) = \{\Delta_1, \dots, \Delta_n\}$ , is the labeled minimal subset bigraph  $G = \langle U, V, E, L \rangle$  such that

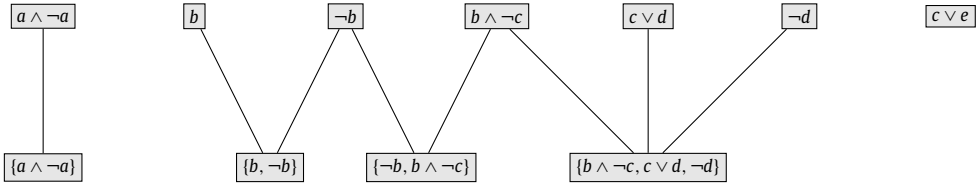


Fig. 1. The labeled syntactic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$ .

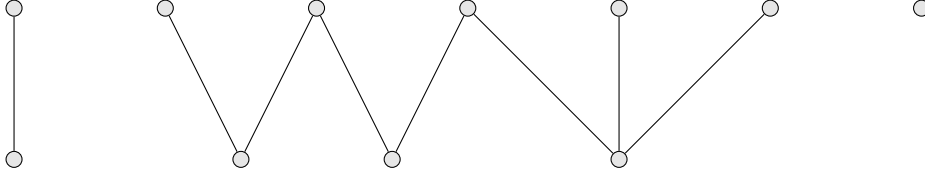


Fig. 2. The syntactic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$ .

- $U = \{u_1, \dots, u_m\}$
- $V = \{v_1, \dots, v_n\}$
- $E = \{(u_i, v_j) \mid \phi_i \in \Delta_j\}$  (every edge corresponds to the set membership relation between a problematic formula  $\phi_i \in K$  and a minimal inconsistent subset  $\Delta_j \in \text{MI}(K)$  such that  $\phi_i \in \Delta_j$ )
- $L : U \cup V \rightarrow \mathcal{L} \cup \mathcal{K} : L(u_i) = \phi_i$  for all  $i, 1 \leq i \leq m$ , and  $L(v_j) = \Delta_j$  for all  $j, 1 \leq j \leq n$ .

In the next example, Fig. 1 shows a labeled syntactic inconsistency bigraph for a propositional knowledgebase.

**Example 1.** Let  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$ . Then  $\text{MI}(K) = \{\{a \wedge \neg a\}, \{b, \neg b\}, \{\neg b, b \wedge \neg c\}, \{b \wedge \neg c, c \vee d, \neg d\}\}$ .

Next, we remove the labels to obtain the syntactic inconsistency bigraph for the  $K$  of Example 1 in Fig. 2.

Our interest is in the unlabeled case; we showed the labeled version only to make things clear. The importance of syntactic inconsistency bigraphs is that they can be used to define a class of inconsistency measures, called syntactic inconsistency measures. That is, for a given knowledgebase  $K$  we construct its syntactic inconsistency bigraph and apply a function on it to obtain the result. In fact, five of the seven inconsistency measures are syntactic inconsistency measures as we now demonstrate.

**Definition 6.** The following are types of syntactic inconsistency measure.

- $I_B(K) = 1$  if  $V \neq \emptyset$  and  $I_B(K) = 0$  if  $V = \emptyset$
- $I_M(K) = |V|$
- $I_P(K) = |u \in U|$  such that  $\text{Deg}(u) \neq 0$
- $I_R(K) = \min\{|U'| \mid U' \subseteq U \text{ and } \forall v \in V \exists u' \in U' \text{ such that } u' \in \text{Adj}(v)\}$
- $I_{P(r)}(K) = \frac{I_P(K)}{|U|}$

$I_a$  and  $I_{a(r)}$  are not syntactic measures because from the syntactic inconsistency bigraph there is no way to tell how many atoms are in the problematic formulas. In the next example, we use the syntactic inconsistency bigraph for the knowledgebase of Example 1 to calculate the values of the five syntactic inconsistency measures.

**Example 2.** Continuing with Example 1 we obtain from the syntactic inconsistency bigraph of Fig. 2 that  $I_B(K) = 1, I_M(K) = 4, I_P(K) = 6, I_R(K) = 3$  and  $I_{P(r)} = \frac{6}{7}$ .

As mentioned earlier, a syntactic inconsistency bigraph is a special case of a minimal subset bigraph. The following important result gives an important completeness result for syntactic inconsistency bigraphs.

**Theorem 1** (From [14]). For every minimal subset bigraph  $G$  there is a knowledgebase  $K$  for which  $G$  is the inconsistency bigraph.

In conclusion, syntactic inconsistency bigraphs provide a general framework for situating many inconsistency measures from the literature, and for identifying new inconsistency measures. However, they do not account for semantic approaches to measuring inconsistency.

**Table 3**  
Truth values needed for all the connectives.

$\alpha$	F	T	B
$\neg\alpha$	?	?	?

$\wedge$	F	T	B
F	?	?	?
T	?	?	?
B	?	?	?

$\vee$	F	T	B
F	?	?	?
T	?	?	?
B	?	?	?

### 4. 3-Valued logics

We would like to think of semantic inconsistency measures as ones that are defined using the models of a knowledgebase. But this does not work in 2-valued logic because an inconsistent knowledgebase does not have a model. As the idea of an inconsistency measure is to evaluate inconsistent knowledgebases, in order to obtain models we must extend the logic in some way. In this paper we use three-valued logics, writing the letter  $B$  for the third truth value. Then we use  $B$  as a *designated* truth value, that is, a truth value that is considered acceptable for models, just like  $T$ . Having two designated truth values out of three, instead of one out of two, enlarges the space of models, allowing inconsistent knowledgebases to have models that can then be used to define semantic inconsistency measures. For the rest of the paper an interpretation  $i$  will be a 3-valued interpretation, that is,  $i : \mathcal{A} \rightarrow \{T, F, B\}$ . For the examples, involving atoms  $a, b, c, d$ , etc., we may also represent an interpretation  $i$  as tuple where the first item is  $i(a)$ , the second item is  $i(b)$ , and so on (e.g.  $(T, F, B, T)$  represents  $i(a) = T, i(b) = F, i(c) = B$ , and  $i(d) = T$ ). In order to extend  $i$  from atoms to formulas we need the three-valued truth tables for negation, conjunction, and disjunction. That is, we need to fill in each entry of the truth tables as shown in Table 3.

The specification of the truth tables results in a logic  $L$ . For every interpretation  $i$  every formula gets assigned a truth value using the truth tables in the standard way. We write  $i_L(\phi)$  for the interpretation of the formula  $\phi$  in  $L$ . We write  $i \models_L \phi$  iff  $i(\phi) \in \{T, B\}$  and say that  $i$  is a **model** of  $\phi$ . For a knowledgebase  $K$ ,  $i$  is a **model** of  $K$  iff  $i \models_L \phi$  for every  $\phi \in K$ . We say that  $K$  is **equivalent** to  $K'$  and write  $K \equiv_L K'$  iff  $K$  and  $K'$  have the same models, that is, for every interpretation  $i$ ,  $i \models_L K$  iff  $i \models_L K'$ . As there are 21 entries to fill in for the truth tables, and each entry can take on one of three values, potentially there are  $3^{21}$  truth tables to consider. However, only a small number of these can be considered a genuine logic. We now show that by requiring some reasonable properties we drastically reduce the number of possibilities. We consider the following properties that constrain the interpretations.

1. **Supraclassical property.** The 3-valued logic extends classical 2-valued logic (i.e. if  $i(\alpha), i(\beta) \in \{T, F\}$ , then  $i(\alpha \wedge \beta), i(\alpha \vee \beta)$ , and  $i(\neg\alpha)$  have the same values as in classical logic).
2. **Double negation property.** The rule of double negation holds (i.e.  $i(\neg\neg\alpha) = i(\alpha)$  holds).
3. **Input-output property.** Conjunction and disjunction result in one of the input values (i.e.  $i(\alpha \wedge \beta), i(\alpha \vee \beta) \in \{i(\alpha), i(\beta)\}$ ).
4. **Commutativity law.** Conjunction and disjunction are commutative (i.e.  $i(\alpha \wedge \beta) = i(\beta \wedge \alpha)$  and  $i(\alpha \vee \beta) = i(\beta \vee \alpha)$ ).
5. **De Morgan's laws.** De Morgan's laws hold (i.e.  $i(\neg(\alpha \wedge \beta)) = i(\neg\alpha \vee \neg\beta)$  and  $i(\neg(\alpha \vee \beta)) = i(\neg\alpha \wedge \neg\beta)$ ).
6. **And-or property.** If a conjunction yields a designated value, so must the corresponding disjunction (i.e. if  $i(\alpha \wedge \beta) \in \{T, B\}$ , then  $i(\alpha \vee \beta) \in \{T, B\}$ ).

We have focused on these properties as they capture some desirable properties for reasoning with inconsistency in applications (such as reasoning with inconsistent requirement specifications in software engineering or reasoning with inconsistent reports in intelligence analysis): supraclassicality ensures that we have classical reasoning when the information is consistent; the double negation property, the commutativity law, and De Morgan's law are valuable for ensuring natural behaviour of the negation, conjunction and disjunction operators; the input-output property ensures that the assignment to a formula is based on the assignment to subformulas; and the and-or property ensures that the assignment to conjunction cannot be better, in terms of designated values, than the assignment to disjunction.

**Theorem 2.** *There are three 3-valued logics whose truth tables satisfy the supraclassical property, the double negation property, the input-output property, the commutativity law, De Morgan's laws, and the and-or property.*

**Proof.** We go through the six conditions one by one showing how each diminishes the number of possible truth tables.

1. **Supraclassical property** This means that the values in the truth tables conform to the ones in Table 1. This fills in 10 entries reducing the number of possibilities to  $3^{11}$ . This is represented by Table 4.
2. **Double negation property** Double negation states that  $\neg\neg Z = Z$  for every truth value  $Z$ . This fills in 1 entry reducing the number of possibilities to  $3^{10}$ . This is represented by Table 5.
3. **Input-output property** This means that, for example,  $T \vee B = F$  is not allowed. In particular it specifies that  $B \wedge B = B \vee B = B$ , which assures that  $i(a \wedge \neg a) = i(a \vee \neg a) = i(a)$ . So this takes care of 2 entries. Furthermore, the other 8 entries are restricted to 2 possible values. This leaves  $2^8$  possibilities. We use  $V_1?V_2$  to indicate that the unknown value must be either  $V_1$  or  $V_2$ . This is represented by Table 6.



**Table 4**  
Truth tables after Step 1.

$\alpha$	F	T	B
$\neg\alpha$	T	F	?

$\wedge$	F	T	B
F	F	F	?
T	F	T	?
B	?	?	?

$\vee$	F	T	B
F	F	T	?
T	T	T	?
B	?	?	?

**Table 5**  
Truth tables after Step 2.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	?
T	F	T	?
B	?	?	?

$\vee$	F	T	B
F	F	T	?
T	T	T	?
B	?	?	?

**Table 6**  
Truth tables after Step 3. Each question mark represents one of the two adjacent truth values.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	F?B
T	F	T	T?B
B	F?B	T?B	B

$\vee$	F	T	B
F	F	T	F?B
T	T	T	T?B
B	F?B	T?B	B

**Table 7**  
Truth tables after Step 4. X is either F or B, Y is either T or B, U is either F or B, and V is either T or B.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	X
T	F	T	Y
B	X	Y	B

$\vee$	F	T	B
F	F	T	U
T	T	T	V
B	U	V	B

4. Commutativity law On account of commutativity, filling in the bottom row automatically fills in the last column. In the following table we use X, Y, U, and V for the possible values, reducing the number of possibilities to  $2^4$ . This is represented by Table 7.
5. De Morgan's Laws The equalities are  $\neg(Z_1 \wedge Z_2) = \neg Z_1 \vee \neg Z_2$  and  $\neg(Z_1 \vee Z_2) = \neg Z_1 \wedge \neg Z_2$ . At this point, omitting the cases for the classical truth values, we can write each truth table in the form XYUV. For example, BTFT means that  $F \wedge B = B$ ,  $T \wedge B = T$ ,  $F \vee B = F$ , and  $T \vee B = T$ . Next, we list the 16 remaining truth tables in alphabetical order.

- 1: BBBB    2: BBBT    3: BBFB    4: BBFT  
 5: BTBB    6: BTBT    7: BTFB    8: BTFT  
 9: FBBB    10: FBBT    11: FBFB    12: FBFT  
 13: FTBB    14: FTBT    15: FTFB    16: FTFT.

To show how we can examine these cases, consider case 2 (BBBT). So  $F \wedge B = X = B$ ,  $T \wedge B = Y = B$ ,  $F \vee B = U = B$ , and  $T \vee B = V = T$ . Focusing on V, we see that  $T \vee B = T$  implies  $\neg(T \vee B) = F$  and focusing on X, we see that  $F \wedge B = B$  implies  $\neg T \wedge \neg B = B$ . So  $\neg(T \vee B) = F \neq \neg T \wedge \neg B = B$ , and hence De Morgan's law is violated.

It turns out that 12 of the 16 cases violate De Morgan's Laws as we now demonstrate by giving an appropriate counterexample that may apply to several at one time.

- 2:, 8:  $\neg(T \vee B) = F \neq \neg T \wedge \neg B = B$   
 3:, 4:  $\neg(T \wedge B) = B \neq \neg T \vee \neg B = F$   
 5:, 6:  $\neg(T \wedge B) = F \neq \neg T \vee \neg B = B$   
 9:, 11:, 13:  $\neg(F \wedge B) = T \neq \neg F \vee \neg B = B$   
 12:, 15:  $\neg(F \vee B) = T \neq \neg F \wedge \neg B = B$   
 14:  $\neg(F \vee B) = B \neq \neg F \wedge \neg B = T$

This leaves 4 possibilities: BBBB, BTFB, FBFT, and FTFT.

6. And-or property We think of designated values as higher than nondesignated values and would like to distinguish between conjunction and disjunction so that conjunction cannot give a higher value than disjunction. This eliminates BTFB where  $F \wedge B = B$  while  $F \vee B = F$ , leaving 3 logics.  $\square$

We assign these three logics the names A-logic, B-logic, and C-logic. That is, using the numbering scheme in the proof, A-logic is 10:FBFT, B-logic is 1:BBBB, and C-logic is 16:FTFT. A-logic was proposed as a logic for indeterminates by Łukasiewicz around 1920 and for paradoxes by Priest [45] as the Logic of Paradox. B-logic was proposed in the 1939 by Bochvar for dealing with paradoxes. See [48] for detailed background and references on these logics. C-logic was proposed by Sobociński [49] and used for conditional events in [54] and [17]. The truth tables for them are given in Tables 8 to 10.

**Table 8**  
The truth tables for A-logic.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	F
T	F	T	B
B	F	B	B

$\vee$	F	T	B
F	F	T	B
T	T	T	T
B	B	T	B

**Table 9**  
The truth tables for B-logic.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	B
T	F	T	B
B	B	B	B

$\vee$	F	T	B
F	F	T	B
T	T	T	B
B	B	B	B

**Table 10**  
The truth tables for C-logic.

$\alpha$	F	T	B
$\neg\alpha$	T	F	B

$\wedge$	F	T	B
F	F	F	F
T	F	T	T
B	F	T	B

$\vee$	F	T	B
F	F	T	F
T	T	T	T
B	F	T	B

**Proposition 1.** In A-logic and B-logic every formula can be converted into Conjunctive Normal Form (CNF) and Disjunctive Normal Form (DNF) in such a way that the truth values are the same for all interpretations.

**Proof.** In addition to the six conditions mentioned in Theorem 2, A-logic and B-logic also satisfy the associative laws:  $Z_1 \wedge (Z_2 \wedge Z_3) = (Z_1 \wedge Z_2) \wedge (Z_1 \wedge Z_3)$  and  $Z_1 \vee (Z_2 \vee Z_3) = (Z_1 \vee Z_2) \vee (Z_1 \vee Z_3)$  and the distributive laws:  $Z_1 \wedge (Z_2 \vee Z_3) = (Z_1 \wedge Z_2) \vee (Z_1 \wedge Z_3)$  and  $Z_1 \vee (Z_2 \wedge Z_3) = (Z_1 \vee Z_2) \wedge (Z_1 \vee Z_3)$ . CNF and DNF can then be constructed for any formula using all these laws of logic the same way as in classical 2-valued logic.  $\square$

The distributive laws do not hold in C-logic. Consider the case where  $a \vee (b \wedge c) \equiv (a \vee b) \wedge (a \vee c)$ . Let  $i(a) = F$ ,  $i(b) = T$ , and  $i(c) = B$ . Then  $i_C(a \vee (b \wedge c)) = F \vee T = T \neq i_C((a \vee b) \wedge (a \vee c)) = T \wedge F = F$  where  $i_C$  denotes the interpretation  $i$  extended to formulae according to C-logic. The same  $i$  can be used to give different truth values to  $a \vee (b \wedge c)$  and  $(a \vee b) \wedge (a \vee c)$ .

Another type of property to consider is identity. For disjunction, identity ensures that if one disjunct is assigned T, then the disjunction is assigned T (i.e. if  $i(\alpha) = T$  or  $i(\beta) = T$ , then  $i(\alpha \vee \beta) = T$ ), and for conjunction, identity ensures if one conjunct is assigned T, then the conjunction takes the truth value of the other conjunct (i.e. if  $i(\alpha) = T$ , then  $i(\alpha \wedge \beta) = i(\beta)$ ). A-logic satisfies both these identities, whereas B-logic only satisfies conjunction identity, and C-logic only satisfies disjunction identity.

We may think of these logics as follows. A-logic uses the ordering of the truth values as  $F < B < T$  where conjunction gives the lower value and disjunction the higher value. In B-logic the truth value B absorbs every other truth value for conjunction and disjunction. In a way, C-logic is the opposite of B-logic: here the B value is ignored for both conjunction and disjunction, as long as the other value is not also B. We think that A-logic has an advantage over the other two logics because it distinguishes conjunction and disjunction when exactly one of the truth values is B. We put B-logic as a better choice than C-logic because it obeys the distributive laws. Nevertheless, all three logics appear appropriate for further consideration.

The three logics offer different options in practice. B-logic is the most skeptical. A disjunction (respectively conjunction) is assigned B whenever a disjunct (respectively conjunct) is assigned B. Hence, a B assignment to a formula is propagated to any formula that contains it as a subformula. The least skeptical is C-logic. Here a disjunction (respectively conjunction) is assigned B only in the case that both disjuncts (respectively conjuncts) are assigned B. So C-logic sets a higher threshold for a formula to be assigned B based on its subformulas.

Next we show that a model of a formula in one of the three logics need not be a model of the same formula in a different logic. Consider the interpretation  $i$  where  $i(a) = F$  and  $i(b) = B$ . Let  $\phi = a \wedge b$ . Then,  $i_C(\phi) = i_A(\phi) = F$ , while  $i_B(\phi) = B$ . Hence  $i$  is a B-model of  $\phi$  but not an A-model or a C-model. Let  $\psi = a \vee b$ . Then,  $i_A(\psi) = i_B(\psi) = B$  while  $i_C(\psi) = F$ . Here,  $i$  is an A-model and B-model of  $\psi$  but not a C-model.

We end this section by giving some connections between these three logics.

**Proposition 2.** For every formula  $\phi$ , (1) if  $i_C(\phi) = B$  then  $i_A(\phi) = B$  and (2) if  $i_A(\phi) = B$  then  $i_B(\phi) = B$ .

**Proof.** We proceed by induction on the number of connectives in the formula. In the base case, where  $\phi$  is an atom, say  $a$ , the result is immediate as  $i_C(a) = i_A(a) = i_B(a)$ . In the inductive step, for both (1) and (2) there are three subcases: (a)  $\phi = \neg\psi$ , (b)  $\phi = \psi_1 \wedge \psi_2$ , and (c)  $\phi = \psi_1 \vee \psi_2$ , where by the inductive hypothesis the condition holds for  $\psi$ ,  $\psi_1$ , and  $\psi_2$ . Case (a) is immediate because of the truth table for negation:  $\neg B = B$ . Thus it suffices to consider subcases (b) and (c) for



(1) and (2). The proofs are the same for cases (b) and (c) for both (1) and (2).

(1) Suppose that  $i_C(\phi) = B$ . Then  $i_C(\phi_1) = i_C(\phi_2) = B$ . Hence  $i_A(\psi_1) = i_A(\psi_2) = B$ . Therefore,  $i_A(\phi) = B$ .

(2) Suppose that  $i_A(\phi) = B$ . Then either  $i_A(\psi_1) = B$  or  $i_A(\psi_2) = B$  (or both). Hence either  $i_B(\psi_1) = B$  or  $i_B(\psi_2) = B$ . For both cases  $i_A(\phi) = B$ .  $\square$

**Theorem 3.** For every formula  $\phi$  and interpretation  $i$ , (1) if  $i \models_C \phi$  then  $i \models_A \phi$  and (2) if  $i \models_A \phi$  then  $i \models_B \phi$ .

**Proof.** By Proposition 2 it suffices to show that (1) if  $i_C(\phi) = T$  then  $i \models_A \phi$  and (2) if  $i_A(\phi) = T$  then  $i \models_B \phi$ .

(1) We prove a stronger result: (a) if  $i_C(\phi) = T$  then  $i_A(\phi) \neq F$  and (b) if  $i_C(\phi) = F$  then  $i_A(\phi) \neq T$  simultaneously by induction on the number of connectives in the formula.

In the base case, where  $\phi$  is an atom, say  $a$ , the result is immediate as  $i_C(a) = i_A(a)$ . In the inductive step, for both (a) and (b) there are three subcases: (i)  $\phi = \neg\psi$ , (ii)  $\phi = \psi_1 \wedge \psi_2$ , and (iii)  $\phi = \psi_1 \vee \psi_2$ , where by the inductive hypothesis the condition holds for  $\psi$ ,  $\psi_1$ , and  $\psi_2$ .

Subcase (i) The inductive hypothesis yields that if  $i_C(\psi) = T$  then  $i_A(\psi) \neq F$  and if  $i_C(\psi) = F$  then  $i_A(\psi) \neq T$ . Suppose that  $i_C(\phi) = T$ . Then  $i_C(\psi) = F$ ; hence  $i_A(\psi) \neq T$ , yielding  $i_C(\phi) \neq F$ . Next, if  $i_C(\phi) = F$  then  $i_C(\psi) = T$ ; hence  $i_A(\psi) \neq F$ , so  $i_A(\phi) \neq T$ .

Subcase (ii) The inductive hypothesis yields that for  $i = 1, 2$  if  $i_C(\psi_i) = T$  then  $i_A(\psi_i) \neq F$  and if  $i_C(\psi_i) = F$  then  $i_A(\psi_i) \neq T$ . Suppose that  $i_C(\phi) = T$ . We may assume that  $i_C(\psi_1) = T$  and ( $i_C(\psi_2) = T$  or  $i_C(\psi_2) = B$ ). Hence  $i_A(\psi_i) \neq F$ . Thus  $i_C(\phi) \neq F$ . Next, if  $i_C(\phi) = F$  then we may assume that  $i_C(\psi_1) = F$ . So  $i_A(\psi_1) = F$  and then  $i_A(\phi) = F \neq T$ .

Subcase (iii) The inductive hypothesis yields that for  $i = 1, 2$  if  $i_C(\psi_i) = T$  then  $i_A(\psi_i) \neq F$  and if  $i_C(\psi_i) = F$  then  $i_A(\psi_i) \neq T$ . Suppose that  $i_C(\phi) = T$ . We may assume that  $i_C(\psi_1) = T$ . Therefore  $i_A(\psi_1) \neq F$ . But this implies that  $i_A(\phi) \neq F$ . Next, if  $i_C(\phi) = F$  we may assume that  $i_C(\psi_1) = F$  and  $i_C(\psi_2) \neq T$ . Hence  $i_A(\psi_1) \neq T$  and  $i_A(\psi_2) \neq T$ . Therefore,  $i_C(\phi) \neq T$ .

(2) A similar reasoning as for (1) can be done. However, there is a shorter proof. Use Proposition 1 to put  $\phi$  into CNF. Hence,  $\phi = C_1 \wedge \dots \wedge C_k$  where  $C_i = \ell_{i1} \vee \dots \vee \ell_{iti}$ . As each  $\ell$  is a literal,  $i_A(\ell) = i_B(\ell)$ . As  $i_A(\phi) = T$ ,  $i_A(C_i) = T$  for all  $1 \leq i \leq k$ . This means that for each  $C_i$  there is some literal in  $C_i$ , say  $\ell_{ij}$  such that  $i(\ell_{ij}) = T$ . Therefore  $i \models_B C_i$  for all  $1 \leq i \leq k$  and so  $i \models_B \phi$ .  $\square$

There is a wide variety of interesting 3-valued logics. For reviews, see [2,10]. However, only some of these logics use the third truth value for a notion of contradiction. By focusing on the six properties considered in Theorem 2, we can concentrate on three 3-valued logics for reasoning with inconsistent information. We will consider other 3-valued logics in the discussion section at the end of the paper.

### 5. Minimal L-atomic subsets

In order to define semantic inconsistency measures we will deal with the models of the three logics we defined previously. We start by specifying a set of atoms  $A = \{a_1, \dots, a_n\}$  for which we define 3-valued interpretations. In general,  $A = \text{Atoms}(K)$  for the knowledgebase under consideration because the interpretation of an atom not in  $\text{Atoms}(K)$  is irrelevant. For what follows it will be convenient to think of  $A$  as a sequence,  $A = (a_1, \dots, a_n)$ , as then we can write an interpretation  $i$  as a sequence of truth values  $(V_1, \dots, V_n)$  where each  $V_j \in \{F, T, B\}$  and  $i(a_j) = V_j$ . Recall that for each logic an interpretation also assigns a truth value to every formula. Let  $S \subseteq A$ .

Next, we introduce a notation that will be useful when we define semantic inconsistency measures. We use  $S$  to define an interpretation based on  $i$  that assigns  $B$  to every atom in  $S$  as follows.

$$i[S](a) = \begin{cases} B & \text{if } a \in S \\ i(a) & \text{otherwise} \end{cases}$$

So the interpretation  $i[S]$  is obtained from the interpretation  $i$  by making the changes specified by the atoms in  $S$ . This notation allows us to consider minimal changes to  $i$  by considering minimal sets  $S$  that make  $i[S]$  into a model. In the rest of this section, we will investigate the nature of these modified interpretations, and then, in subsequent sections, we will use the notation for defining semantic inconsistency measures.

Our interest is in knowledgebases but it is easier to work with single formulas. It turns out that for A-logic and C-logic that is all that is needed.

**Proposition 3.** For  $L \in \{A, C\}$ ,  $i \models_L K$  where  $K = \{\phi_1, \dots, \phi_m\}$  iff  $i \models_L \phi_1 \wedge \dots \wedge \phi_m$ .

**Proof.** ( $\rightarrow$ ) By definition,  $i \models_L K$  iff  $i \models_L \phi_j$  for  $1 \leq j \leq m$ . Thus,  $i_L(\phi_j) \in \{B, T\}$  for  $1 \leq j \leq m$ . By the truth tables for A-logic and C-logic,  $i \models_L \phi_1 \wedge \dots \wedge \phi_m$ .

( $\leftarrow$ ) Suppose that  $i \models_L \phi_1 \wedge \dots \wedge \phi_m$ , that is,  $i_L(\phi_1 \wedge \dots \wedge \phi_m) \in \{T, B\}$ . By the truth tables for A-logic and C-logic,  $i_L(\phi_j) = F$  is not possible for any  $j$ ,  $1 \leq j \leq m$ . Hence  $i_L(\phi_j) \in \{T, B\}$  for all  $j$ ,  $1 \leq j \leq m$ , and so  $i \models_L \phi_j$  for all  $j$ ,  $1 \leq j \leq m$ , which means that  $i \models_L K$ .  $\square$

This proposition does not apply to B-logic. The reason is that in B-logic  $B \wedge F = B$ . For example, let  $K = \{a, b\}$  and  $i(a) = B$  and  $i(b) = F$ . Here,  $i \not\models_B K$  but  $i \models_B a \wedge b$ . So the proposition means that for A-logic and C-logic it suffices to deal

with the models of a single formula but for B-logic we must deal with the models of a set of formulas. In the following we will usually deal with a single formula and explain what else needs to be done for B-logic.

If a knowledgebase  $K$  is inconsistent, there is no  $i$  such that  $i \models_L K$  and for each atom  $a$ ,  $i(a) = T$  or  $i(a) = F$ , because each logic  $L$  is an extension of classical 2-valued logic. However, there is always an interpretation  $i[S]$  such that  $i[S] \models_L K$  for each logic. We can always choose  $S = A$  because if every atom gets the value  $B$ , every formula also gets the value  $B$ . But, in general, subsets of  $A$  smaller than  $S$  suffice for this purpose.

**Example 3.** Consider  $K = \{a, \neg a \vee \neg b, b, c\}$ . Also consider the following two valued interpretations:  $i_1 = (T, T, T)$ ,  $i_2 = (T, T, F)$ ,  $i_3 = (T, F, T)$ ,  $i_4 = (T, F, F)$ ,  $i_5 = (F, T, T)$ ,  $i_6 = (F, T, F)$ ,  $i_7 = (F, F, T)$ ,  $i_8 = (F, F, F)$ . So for all  $L$ , we have the following:

- If  $S \in \{\emptyset, \{c\}\}$ , then there is no  $i \in \{i_1, \dots, i_8\}$  such that  $i[S] \models_L K$ ;
- If  $S$  is  $\{a\}$ , and  $i \in \{i_1, i_5\}$ , then  $i[S] \models_L K$ ;
- If  $S$  is  $\{b\}$ , and  $i \in \{i_1, i_3\}$ , then  $i[S] \models_L K$ ;
- If  $S$  is  $\{a, b\}$ , and  $i \in \{i_1, i_3, i_5, i_7\}$ , then  $i[S] \models_L K$ ;
- If  $S$  is  $\{a, c\}$ , and  $i \in \{i_1, i_2, i_5, i_6\}$ , then  $i[S] \models_L K$ ;
- If  $S$  is  $\{b, c\}$ , and  $i \in \{i_1, i_2, i_3, i_4\}$ , then  $i[S] \models_L K$ ;
- If  $S$  is  $\{a, b, c\}$ , and  $i \in \{i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8\}$ , then  $i[S] \models_L K$ .

Whenever for some classical 2-valued interpretation  $i$ ,  $i[S] \models_L K$ , we say that  $S$  is an **L-atomic subset** for  $K$ . In particular, if  $K$  is consistent, there is always some classical interpretation  $i$  for which  $\emptyset$  is an L-atomic subset. The following result shows a connection among the L-atomic subsets of the three logics. When we introduce the inconsistency measures later, we will see how this result gives insights into the relationships among the measures.

**Proposition 4.** (1) Every C-atomic subset of a knowledgebase  $K$  is an A-atomic subset of  $K$ . (2) Every A-atomic subset of a knowledgebase  $K$  is a B-atomic subset of  $K$ .

**Proof.** Immediate from Theorem 3.  $\square$

The following result allows us to deal only with some of the L-atomic subsets. Our motivation for investigating this is that we want to focus on minimal L-atomic subsets which we define below.

**Theorem 4.** Let  $K$  be a knowledgebase and  $i$  a classical 2-valued interpretation.

(1) If  $i[S] \models_A K$  and  $S \subseteq S'$  then  $i[S'] \models_A K$ . (2) If  $i[S] \models_B K$  and  $S \subseteq S'$  then  $i[S'] \models_B K$ .

**Proof.** By Proposition 3 for A-logic it suffices to deal with a single formula.

(1) By Proposition 1 we may assume that  $i[S] \models_A \phi$  where  $\phi$  is in DNF. So,  $\phi = D_1 \vee \dots \vee D_k$  where each  $D_j = \ell_{j1} \wedge \dots \wedge \ell_{jt_j}$  and each  $\ell$  is a literal. We get that  $i[S] \models_A D_j$  for some  $j$ ,  $1 \leq j \leq k$  and then  $i[S] \models_A \ell_{jt}$  for all literals in  $D_j$ . Then, for  $S'$  where  $S \subseteq S'$ , some additional atoms may get the truth value  $B$ . Such a change does not change things in the sense that  $i[S'] \models_A \ell_j$  for each literal in  $D_j$ . Therefore,  $i[S'] \models_A \phi$ .

(2) In this case we must deal with the whole knowledgebase  $K$ . Let  $K = \{\phi_1, \dots, \phi_n\}$ . Consider any  $\phi_t$  for  $1 \leq t \leq n$ . As  $i[S] \models_B K$ ,  $i[S] \models_B \phi_t$  also. We will show that  $i[S'] \models_B \phi_t$ . By Proposition 1 we may assume that  $\phi_t$  is in DNF, so that  $\phi_t = D_1 \vee \dots \vee D_k$  where  $D_j = \ell_{j1} \wedge \dots \wedge \ell_{jt_j}$  and each  $\ell$  is a literal. The rest of the proof follows the same reasoning as for (1).  $\square$

This theorem does not work for C-logic. Let  $\phi = a \vee b$  and consider the 2-valued interpretation  $i(a) = T$  and  $i(b) = F$ . Then  $i[\emptyset] \models_C \phi$ . So here  $S = \emptyset$ . But for  $S' = \{a\}$ ,  $i[S'] \not\models_C \phi$  because in C-logic  $B \vee F = F$ .

**Definition 7.** We say that  $S$  is a **minimal L-atomic subset** of  $K$  iff  $S$  is an L-atomic subset of  $K$  and there is no classical 2-valued interpretation  $i'$  and set  $S' \subset S$  such that  $i'[S'] \models_L K$ . We write  $MAS_L(K)$  for the set of all minimal L-atomic subsets of  $K$ .

These concepts were defined originally for A-Logic in [41]. Specifically, a minimal A-atomic subset of  $K$  is written as  $\omega!(K)$  for some  $\omega \in \text{MinMod}_{LP}(K)$ . Then,  $MAS_A(K)$  is called  $BA(K)$  there.

By Theorem 4, in the case of A-logic and B-logic it suffices to know the minimal L-atomic subsets to get all L-atomic subsets. Although this result does not hold for C-logic, minimal L-atomic subsets are a convenient way to study the semantics of inconsistencies for all three logics.

The following result shows a connection among the minimal L-atomic subsets of the three logics.

**Proposition 5.** (1) If  $S \in MAS_C(K)$  then there is an  $S' \subseteq S$  such that  $S' \in MAS_A(K)$ . (2) If  $S \in MAS_A(K)$  then there is an  $S' \subseteq S$  such that  $S' \in MAS_B(K)$ .

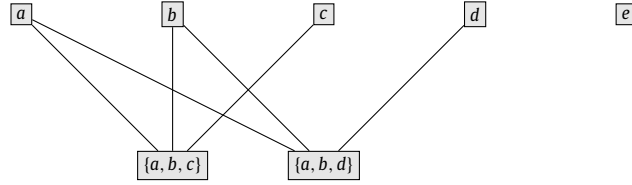


Fig. 3. The labeled semantic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in A-logic.

**Proof.** (1) If  $S \in MAS_C(K)$  then, by Proposition 4 (1)  $S$  is a C-atomic subset of  $K$ . Hence, either  $S \in MAS_A(K)$  or there is an  $S' \subseteq S$  such that  $S' \in MAS_A(K)$ .  
 (2) Same as (1) with the substitution of “A” for “C” and “B” for “A” using Proposition 4 (2).  $\square$

**Example 4.** This example uses an iceberg inconsistency (see [15]). Let  $K_1 = \{a \wedge \neg a \wedge b, \neg b\}$ . From the syntactic point of view,  $\neg b$  is a free formula and so is mostly irrelevant for calculating a syntactic inconsistency measure except if the total number of formulas is used. This misses the fact that there is a hidden inconsistency between the two formulas. In fact,  $\neg b$  is quite relevant from the semantic point of view.  $K_1$  has one minimal C-atomic subset,  $S = \{a, b\}$  which is also the single minimal A-atomic subset. However,  $K_1$  has 2 minimal B-atomic subsets:  $\{a\}$  and  $\{b\}$ . In both cases we can use  $i = (T, F)$ .

**Example 5.** This example is a slight modification of the previous example. Let  $K_2 = \{a \wedge \neg a \wedge b, \neg b \vee (c \wedge \neg c)\}$ . Here also the second formula is free.  $K_2$  has one minimal C-atomic subset,  $S = \{a, b, c\}$ . But  $K_2$  has 2 minimal A-atomic subsets,  $\{a, b\}$  and  $\{a, c\}$  using the interpretations  $i_1 = (T, F, T)$  and  $i_2 = (T, T, T)$  respectively. Finally,  $K_2$  has the same 2 minimal B-atomic subsets as  $K_1$ , namely  $\{a\}$  and  $\{b\}$  using the interpretation  $i = (T, F, T)$  for both.

In conclusion, we have introduced the notion of a subset of atoms in the language being an L-minimal subset of a knowledgebase  $K$ . We will use this notion in the rest of this paper to allow us to focus on where inconsistencies lie in the knowledgebase  $K$ , and thereby use it as the basis of analysing  $K$ .

### 6. Semantic inconsistency bigraphs

As we explained earlier, the aim of measuring inconsistency is to obtain an evaluation of the imperfections in a set of formulas. This evaluation may then be used to help decide on some course of action, such as rejecting some of the formulas, resolving the inconsistency, seeking better sources of information, etc. Many inconsistency measures have been proposed for this purpose. Each has its rationale. But given the diversity of measures, it is beneficial to delineate the space of options for measures and to classify measures systematically.

In Section 3, we reviewed previous work on syntactic inconsistency bigraphs. These are a special case of minimal subset bigraphs where the set is the formulas of the knowledgebase and the property for a subset is that it is inconsistent. Syntactic inconsistency measures are functions on such bigraphs. In this section we show that an analogous definition can be used to define semantic inconsistency bigraphs. In the next section we will show how to use these bigraphs to define semantic inconsistency measures.

In the following definition, we use a bigraph to represent both the atoms in the formulae in a knowledgebase and the nature of the inconsistencies in terms of the minimal L-atomic subsets. While for clarity we define both labeled and unlabeled semantic inconsistency bigraphs, later we will use only the unlabeled versions. By using the unlabeled version, we will abstract away from the specific atoms and focus on the structure of the bigraphs that will then be used to define the semantic inconsistency measures.

**Definition 8.** The **labeled semantic inconsistency bigraph** for KB  $K$  and logic  $L$  where  $Atoms(K) = \{a_1, \dots, a_m\}$  and  $MAS_L(K) = \{S_1, \dots, S_n\}$ , is the labeled minimal subset bigraph  $G = (U, V, E, L)$  such that

- $U = \{u_1, \dots, u_m\}$
- $V = \{v_1, \dots, v_n\}$
- $E = \{(u_i, v_j) \mid a_i \in S_j\}$  (every edge corresponds to the set membership relation between an atom  $a_i \in Atoms(K)$  and a minimal L-atomic subset  $S_j \in MAS_L(K)$  such that  $a_i \in S_j$ )
- $L : U \cup V \rightarrow Atoms(K) \cup MAS_L(K) : L(u_i) = a_i$  for all  $i, 1 \leq i \leq m$ , and  $L(v_j) = S_j$  for all  $j, 1 \leq j \leq n$ .

**Example 6.** Consider the knowledgebase  $K$  from Example 1:  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$ . Fig. 1 is the labeled syntactic inconsistency bigraph of  $K$ . We obtain  $Atoms(K) = \{a, b, c, d, e, f\}$ . The three logics yield different minimal atomic subsets:  $MAS_A(K) = \{\{a, b, c\}, \{a, b, d\}\}$ ,  $MAS_B(K) = \{\{a, b\}\}$ , and  $MAS_C(K) = \{\{a, b, c, d\}\}$ . So for A-logic, either the atoms  $\{a, b, c\}$  are in conflict or the atoms  $\{a, b, d\}$  are in conflict; whereas in B-logic, only the atoms  $\{a, b\}$  are in conflict; and in C-logic, all the atoms in  $\{a, b, c, d\}$  are in conflict. We show the labeled semantic inconsistency bigraph in Fig. 3 for A-logic.

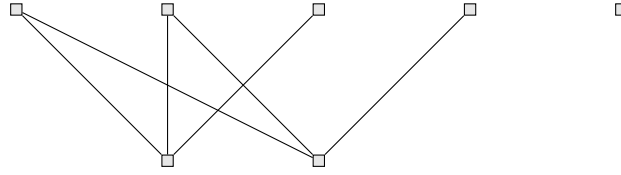


Fig. 4. The semantic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in A-logic.

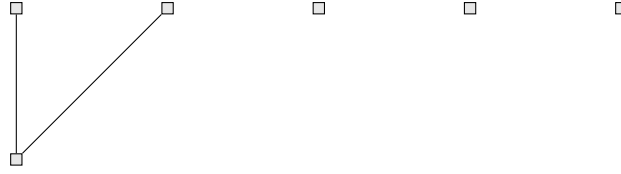


Fig. 5. The semantic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in B-logic.

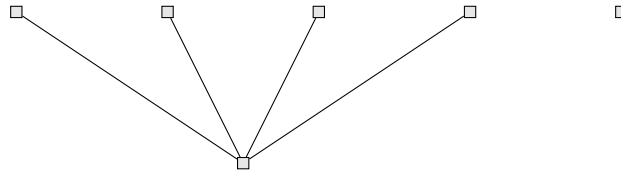


Fig. 6. The semantic inconsistency bigraph for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in C-logic.

Just as in the case of syntactic inconsistency bigraphs, when we remove the labels now we obtain the semantic inconsistency bigraphs. The next three figures represent the semantic inconsistency bigraphs for the three logics.

We mentioned in Section 3 that every minimal subset bigraph represents the syntactic inconsistency bigraph for some knowledgebase. We now consider the same issue for the semantic inconsistency bigraphs for our three logics. It turns out that the same result holds for A-logic and B-logic but not for C-logic.

**Theorem 5.** For every minimal subset bigraph  $G$  there is a knowledgebase whose semantic inconsistency bigraph for A-logic is  $G$ .

**Proof.** Let  $G = \langle U, V, E \rangle$  be a minimal subset bigraph. If  $U = \emptyset$ , set  $K = \emptyset$ . Otherwise, let  $U = \{u_1, \dots, u_m\}$  and  $V = \{v_1, \dots, v_n\}$ . Using  $U$ , set  $\text{Atoms}(K) = \{a_1, \dots, a_m\}$ . For each  $u_i$  such that  $\text{Adj}(u_i) = \emptyset$ , make the atom  $a_j$  a formula of  $K$ . Finally, construct a formula in DNF of the form  $D_1 \vee \dots \vee D_n$  and add it to  $K$  as follows. For each  $v_j$ , let  $\text{Adj}(v_j) = \{u_{j_1}, \dots, u_{j_k}\}$ . Set  $D_j = \{a_{j_1} \wedge \neg a_{j_1} \wedge \dots \wedge a_{j_k} \wedge \neg a_{j_k}\}$ .  $K$  will then have  $G$  as its semantic inconsistency bigraph for A-logic.  $\square$

**Example 7.** Consider the minimal subset bigraph of Fig. 4. The construction yields  $K = \{a_5, (a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2 \wedge a_3 \wedge \neg a_3) \vee (a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2 \wedge a_4 \wedge \neg a_4)\}$ .

**Theorem 6.** For every minimal subset bigraph  $G$  there is a knowledgebase whose semantic inconsistency bigraph for B-logic is  $G$ .

**Proof.** Let  $G = \langle U, V, E \rangle$  be a minimal subset bigraph. If  $U = \emptyset$ , set  $K = \emptyset$ . Otherwise, let  $U = \{u_1, \dots, u_m\}$  and  $V = \{v_1, \dots, v_n\}$ . Using  $U$ , set  $\text{Atoms}(K) = \{a_1, \dots, a_m\}$ . For each  $u_i$  such that  $\text{Adj}(u_i) = \emptyset$ , make the atom  $a_j$  a formula of  $K$ . Find all sets  $T_k$  that are minimal with respect to the property that  $T_k \cap \text{Adj}(v_j) \neq \emptyset$  for all  $1 \leq j \leq n$ . Obtain the sets  $T_1, \dots, T_\ell$ . Thus each  $T_k \subseteq \text{Atoms}(K)$ . Suppose that  $T_k = \{u_{k1}, \dots, u_{kr}\}$ . Set  $K = \{\phi_1, \dots, \phi_\ell\}$  where  $\phi_k = a_{k1} \wedge \neg a_{k1} \wedge \dots \wedge a_{kr} \wedge \neg a_{kr}$ .  $K$  will then have  $G$  as its semantic inconsistency bigraph for B-logic.  $\square$

**Example 8.** Consider the minimal subset bigraph of Fig. 5. Here,  $U = \{u_1, \dots, u_5\}$  and  $\text{Adj}(v_1) = \{u_1, u_2\}$ . Thus,  $T_1 = \{u_1\}$  and  $T_2 = \{u_2\}$ . So,  $K = \{a_1 \wedge \neg a_1, a_2 \wedge \neg a_2, a_3, a_4, a_5\}$

**Example 9.** In Example 7 we constructed a knowledgebase whose semantic inconsistency bigraph for A-logic was Fig. 4. We now use the same figure to construct a knowledgebase for B-logic. Here  $U = \{u_1, \dots, u_5\}$ ,  $\text{Adj}(v_1) = \{u_1, u_2, u_3\}$ , and  $\text{Adj}(v_2) = \{u_1, u_2, u_4\}$ . Thus,  $T_1 = \{u_1\}$ ,  $T_2 = \{u_2\}$ , and  $T_3 = \{u_3, u_4\}$ . So,  $K = \{a_1 \wedge \neg a_1, a_2 \wedge \neg a_2, a_3 \wedge \neg a_3 \wedge a_4 \wedge \neg a_4, a_5\}$ .

Finally, we get to C-logic.

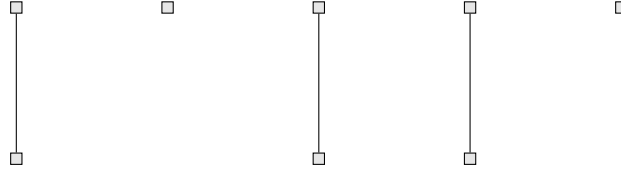


Fig. 7. The semantic inconsistency bigraph for  $K = \{\phi\}$  where  $\phi = (a \wedge \neg a \wedge b) \vee (\neg b \wedge c \wedge \neg c) \vee (a \wedge \neg b \wedge d \wedge \neg d \wedge e)$  in A-logic.

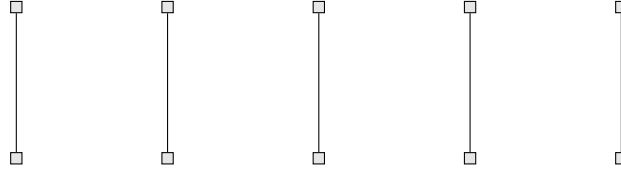


Fig. 8. The semantic inconsistency bigraph for  $K = \{\phi\}$  where  $\phi = (a \wedge \neg a \wedge b) \vee (\neg b \wedge c \wedge \neg c) \vee (a \wedge \neg b \wedge d \wedge \neg d \wedge e)$  in B-logic.

**Theorem 7.** A minimal subset bigraph is the semantic inconsistency bigraph for some knowledgebase in C-logic iff it satisfies at least one of the following conditions: 1)  $G = \emptyset$ , 2) it has at least one element  $u_i \in U$  such that  $\text{Deg}(u_i) = 0$ , 3)  $|V| = 1$ .

**Proof.** We show how to construct a knowledgebase for each of these three cases and why otherwise no such knowledgebase exists.

- 1) Let  $K = \emptyset$ .
- 2) Let  $G = \langle U, V, E \rangle$  be a minimal subset bigraph where  $U = \{u_1, \dots, u_m\}$ ,  $V = \{v_1, \dots, v_n\}$ , and  $\text{Deg}(u_m) = 0$ . Using  $U$ , set  $\text{Atoms}(K) = \{a_1, \dots, a_m\}$ . For each  $u_i$  such that  $\text{Deg}(u_i) = 0$ , make the atom  $a_j$  a formula of  $K$ . In particular,  $a_m$  is such an atom. Then, construct a formula of the form  $D_1 \vee \dots \vee D_n$  and add it to  $K$  as follows. For each  $v_j$ , let  $\text{Adj}(v_j) = \{u_{j1}, \dots, u_{jk}\}$ . Set  $D_j = a_{j1} \wedge \neg a_{j1} \wedge \dots \wedge a_{jk} \wedge \neg a_{jk} \wedge a_m$ .  $K$  will then have  $G$  as its semantic inconsistency bigraph for C-logic. The reason for using  $a_m$  in each disjunct,  $D_j$ , is that an interpretation  $i$  such that  $i(a_m) = T$  and  $i(a_{jk}) = B$  for all  $a_{jk} \in D_j$  makes  $i(D_j) = T$ .
- 3) We must construct a knowledgebase for a minimal subset bigraph  $G = \langle U, V, E \rangle$  where  $U \neq \emptyset$  and for every  $u \in U$ ,  $\text{Deg}(u) > 0$ . This condition means that  $K$  must be inconsistent. By Proposition 3 we can restrict the construction to a single formula  $\phi$  where  $\text{Atoms}(\phi) = \{a_1, \dots, a_m\}$ . As C-logic obeys De Morgan's Laws, we may assume that all negations are applied to atoms. So the formula must contain only conjunctions and disjunctions of literals. Now we proceed to characterize such a formula  $\phi$ . First of all, there cannot be any atoms that appear in  $\phi$  only as positive atoms or only as negated atoms, because for them in  $G$ , there would be a  $u$  such that  $\text{Deg}(u) = 0$ . Consider now what it would mean if  $|V| = 2$ . Starting the construction of such a formula, there must be formulas  $\psi_1$  and  $\psi_2$  such that for their semantic inconsistency bigraphs,  $G_1 = \langle U_1, V_1, E_1 \rangle$  and  $G_2 = \langle U_2, V_2, E_2 \rangle$ ,  $|V_1| = 1$ , and  $|V_2| = 1$ . Therefore,  $\phi$  must be either  $\psi_1 \wedge \psi_2$  or  $\psi_1 \vee \psi_2$ . In either case, we must set all the atoms for both  $\psi_1$  and  $\psi_2$  to  $B$  because otherwise we would have either  $B \wedge F$  or  $B \vee F$  or  $F \wedge B$  or  $F \vee B$ , all of which in C-logic yield  $F$ . Thus, no matter how complex is the formula  $\phi$ , for its semantic inconsistency graph  $G$ ,  $|V| = 1$ . Then, for  $G = \langle \{u_1, \dots, u_m\}, \{v_1\}, \{(u_1, v_1), \dots, (u_m, v_1)\} \rangle$  the corresponding formula is  $\phi = a_1 \wedge \neg a_1 \wedge \dots \wedge a_m \wedge \neg a_m$ .  $\square$

**Example 10.** Consider the minimal inconsistency graph of Fig. 3. We construct a knowledgebase  $K$  for which this graph is the semantic inconsistency bigraph for C-logic. This is an example of Case 2). We obtain  $K = \{(a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2 \wedge a_3 \wedge \neg a_3 \wedge a_5) \vee (a_1 \wedge \neg a_1 \wedge a_2 \wedge \neg a_2 \wedge a_4 \wedge \neg a_4 \wedge a_5)\}$ .

**Example 11.** Let  $K = \{\phi\}$  where  $\phi = (a \wedge \neg a \wedge b) \vee (\neg b \wedge c \wedge \neg c) \vee (a \wedge \neg b \wedge d \wedge \neg d \wedge e)$ . In this case we start with a classical interpretation  $i$  where  $i(e) = T$ . We must set  $S = \{a, b, c, d\}$ . Hence the semantic inconsistency bigraph for C-logic is the same as the one in Fig. 6. We show the semantic inconsistency bigraphs for A-logic and B-logic in Figs. 7 and 8 respectively.

We obtain the following result concerning the various semantic inconsistency bigraphs for a knowledgebase.

**Proposition 6.** Let  $K$  be a knowledgebase whose semantic inconsistency bigraph for L-logic is  $G_L = \langle U_L, V_L, E_L \rangle$  for  $L \in \{A, B, C\}$ . Then

- (1) If  $v \in V_C$  then there is  $v' \in V_A$  such that  $\text{Adj}(v') \subseteq \text{Adj}(v)$ .
- (2) If  $v \in V_A$  then there is  $v' \in V_B$  such that  $\text{Adj}(v') \subseteq \text{Adj}(v)$ .

**Proof.** This follows from Proposition 5 and the construction of the semantic inconsistency bigraphs.  $\square$

In Section 4 we gave a reason to give a preference to A-logic over B-logic and C-logic. The issue was that for B values A-logic distinguishes  $\wedge$  and  $\vee$ , while B-logic and C-logic do not. From this section we see an advantage of B-logic (and A-logic) over C-logic in the sense that it has a vastly larger range of semantic inconsistency bigraphs, and therefore it is more discriminating. Nonetheless, we see that each of these logics has a role in analysing inconsistency, and in the next section we will investigate inconsistency measures based on each.

### 7. Definition and examples of semantic inconsistency measures

We start by defining the concept of semantic measure using semantic inconsistency bigraphs.

**Definition 9.** A **semantic inconsistency measure** of a knowledgebase  $K$  for logic  $L$  is a function that is applied to the semantic inconsistency bigraph of  $K$  for  $L$  that satisfies Definition 1.

This definition is analogous to the definition of a syntactic inconsistency measure which is a function on the syntactic inconsistency bigraph of the knowledgebase as defined in Section 3. For a given function on the syntactic inconsistency bigraph there is a unique syntactic inconsistency measure. But as the semantic inconsistency bigraphs are, in general, different for different logics, a unique semantic inconsistency measure requires the identification of the logic in addition to the function. The following is a basic result about semantic inconsistency measures.

**Proposition 7.** If  $K_1 \equiv_L K_2$  then for any semantic inconsistency measure  $I$  for  $L$ ,  $I(K_1) = I(K_2)$ .

**Proof.**  $K_1 \equiv K_2$  means that  $K_1$  and  $K_2$  have the same models for  $L$ . Thus, they have the same minimal  $L$ -atomic subsets and hence the same semantic bigraphs for  $L$ .  $\square$

Next, we define several semantic inconsistency measures. We will indicate the logic as a superscript, for instance,  $I_x^L$  means  $I_x$  for logic  $L$ . First we note that the syntactic inconsistency measure  $I_B$  (see Definition 1) is also a semantic inconsistency measure with the same definition for all logics  $L$ . Hence we will not deal with  $I_B$  separately as a semantic inconsistency measure. The important question is which functions on the semantic inconsistency bigraphs are intuitively good inconsistency measures. Here we define three functions, thereby getting nine inconsistency measures.

**Definition 10.** The following three functions define semantic inconsistency measures:

- $I_p^L(K) = |\{u \in U \mid \text{Deg}(u) > 0\}|$
- $I_{p(r)}^L(K) = 0$  if  $V = \emptyset$  and  $\frac{I_p^L(K)}{|U|}$  otherwise
- $I_R^L(K) = 0$  if  $V = \emptyset$  and  $\min\{\text{Deg}(v) \mid v \in V\}$  otherwise

$I_p^L$  is the semantic counterpart of the syntactic problematic inconsistency measure,  $I_p$ . But while  $I_p$  counts the number of problematic formulas, that is, the ones that appear in some minimal inconsistent subset,  $I_p^L$  counts the number of *problematic* atoms, that is, the ones that appear in some minimal atomic subset of  $K$  for the logic  $L$ . The relativized version of  $I_p^L$ ,  $I_{p(r)}^L$ , is the ratio of the problematic atoms and the total number of atoms. Finally,  $I_R^L$  is the semantic counterpart of the syntactic repair inconsistency measure,  $I_R$ . But while  $I_R$  counts the minimal number of formulas that must be deleted to make the knowledgebase consistent,  $I_R^L$  counts the minimal size of the set of atoms  $S$  that must be given the value  $B$  such that for some 2-valued interpretation  $i$ ,  $i[S] \models_L K$ . The contension measure, originally defined in [32], and usually written as  $I_C$ , which is usually given as the main example of a semantic inconsistency measure, turns out to be  $I_R^A$  using our notation. We think of it as the semantic repair measure for A-logic.

We did not include a semantic counterpart for  $I_M$  which counts the number of minimal inconsistent subsets. This is  $I_M^L(K) = |V|$  and counts the number of different minimal sets of atoms whose change for a 2-valued interpretation to the value  $B$  resolves the inconsistency of the knowledgebase for logic  $L$ . In other words, it counts the number of alternatives we have for repairs, but it does not reflect how many repairs (in contrast to counting the number of minimal inconsistent subsets), and so we think  $I_R^L(K)$  defined above is more useful for considering repairs. As  $I_M^L$  does not seem to be as useful in the semantic case as its syntactic version, we do not consider it further in this paper. Nonetheless,  $I_M^L(K) = |V|$  may reflect the complexity of resolving the inconsistencies; as the size of  $V$  increases, the difficulty of how to make that selection increases. We will investigate this alternative measure in further work.

Consider now the difference between  $I_a$  given in Definition 1 and  $I_p^L$ . Recall that  $I_a$  counts the number of atoms in the problematic formulas. The problem with  $I_a$  is that problematic formulas may contain many atoms that cause no problems at all. Let  $K = \{a \wedge b \wedge c \wedge d, \neg a \wedge e\}$ . Both formulas are problematic; hence  $I_a(K) = 5$ . But really, only one atom causes the problem, namely  $a$ . Hence  $I_p^L(K) = 1$  for each logic we considered. Intuitively, this seems to be a better way to measure the number of problematic atoms. There is also a big difference for the relative case where  $I_{a(r)}(K) = 1$  while  $I_{p(r)}^L = 0.2$  for all these logics.



Next, we revisit some examples we considered earlier and compute the nine semantic inconsistency measures given in Definition 10 for them.

**Example 12.** We continue with Example 6 where we constructed the semantic inconsistency bigraphs for  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  and obtained graphs in all cases with  $U = \{u_1, \dots, u_6\}$ . For A-logic  $|V| = 2$ , while for B-logic and C-logic  $|V| = 1$ . We obtain the following values for the measures.

	$L = A$	$L = B$	$L = C$
$I_P^L$	4	2	4
$I_{P(r)}^L$	$\frac{4}{5}$	$\frac{2}{5}$	$\frac{4}{5}$
$I_R^L$	3	2	4

**Example 13.** We continue with Example 11 where  $K = \{(a \wedge \neg a \wedge b) \vee (\neg b \wedge c \wedge \neg c) \vee (a \wedge \neg b \wedge d \wedge \neg d \wedge e)\}$  and the semantic inconsistency graph for C-logic is given in Fig. 6. We draw the semantic inconsistency graphs for A-logic and B-logic in Figs. 7 and 8 respectively. We obtain the following values for the measures.

	$L = A$	$L = B$	$L = C$
$I_P^L$	3	5	4
$I_{P(r)}^L$	$\frac{3}{5}$	1	$\frac{4}{5}$
$I_R^L$	1	1	4

We now consider an example involving analysing conflicting witness reports in order to indicate how these measures might be useful in practice.

**Example 14.** Consider the following example taken from [15]. The police is investigating a robbery on a jewellery shop that occurred on a weekday, during working hours. The investigators have taken testimony from all employees who were working on the day of the crime. The witnesses' statements include the following:

- Salesperson: "I did not open the safe, and the criminals carried no guns!"
- Head of security: "Only the manager or the salesperson could have opened the safe, and the criminals carried guns."
- Shop manager: "I did not open the safe."

As the police conceives the possibility of some of the employees having been complicit, they look for contradictions among the versions given. Inconsistent testimonies would imply some witnesses are lying, raising suspicions of complicity against them. The head of security and the salesperson are clearly contradicting each other, but is the manager involved in some contradiction? From the statements above, can one infer that it is possible that the manager is lying?

Consider the following atomic propositions:  $s$  stands for the salesperson opened the safe,  $m$  stands for the manager opened the safe, and  $g$  stands for the criminals carried guns. Now suppose we have the following testimonies.

- Salesperson:  $\phi = \neg s \wedge \neg g$
- Head of security:  $\psi = (s \vee m) \wedge g$
- Store manager:  $\theta = \neg m$

From these testimonies, we can consider the knowledgebase  $\{\phi, \psi, \theta\}$ , whose semantic inconsistency bigraphs are drawn in Figs. 9 to 11. Then we calculate the following inconsistencies measures.

	$L = A$	$L = B$	$L = C$
$I_P^L$	3	3	3
$I_{P(r)}^L$	1	1	1
$I_R^L$	2	1	3

From these measures, we can see that irrespective of which logic is chosen, the number of problematic atoms, namely 3, is the same. But in terms of repairs, the values are different for the three logics with B-logic giving the lowest number of atoms requiring repair, while C-logic gives the highest number of atoms requiring repair.

For comparison, if we use a syntactic inconsistency measure that is based on minimal inconsistent subsets, such as  $I_M$ , then the analysis is based on the subset  $\{\phi, \psi\}$ ; so  $\theta$  is not considered. Hence the potential for the atom  $m$  to be problematic is ignored. In contrast, using a semantic measure allows us to analyse the potential for  $m$  to be problematic, and thereby recognize that the store manager should also be considered under suspicion.

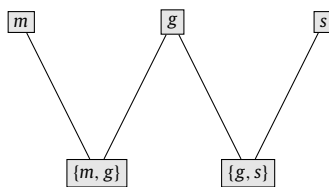


Fig. 9. The labeled semantic inconsistency bigraph for the knowledgebase from Example 14 where  $K = \{\neg s \wedge \neg g, (s \vee m) \wedge g, \neg m\}$  in A-logic.

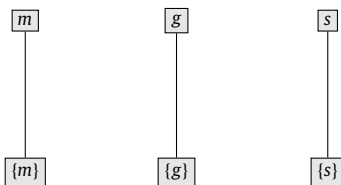


Fig. 10. The labeled semantic inconsistency bigraph for the knowledgebase from Example 14 where  $K = \{\neg s \wedge \neg g, (s \vee m) \wedge g, \neg m\}$  in B-logic.

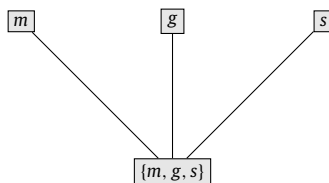


Fig. 11. The labeled semantic inconsistency bigraph for the knowledgebase from Example 14 where  $K = \{\neg s \wedge \neg g, (s \vee m) \wedge g, \neg m\}$  in C-logic.

Table 11

Status of postulates for semantic inconsistency measures.

Inconsistency measures	$I_P^A$	$I_{P(r)}^A$	$I_R^A$	$I_P^B$	$I_{P(r)}^B$	$I_R^B$	$I_P^C$	$I_{P(r)}^C$	$I_R^C$
Monotony	✓	✗	✓	✓	✗	✓	✓	✗	✓
Free-Formula Independence	✗	✗	✗	✓	✗	✓	✗	✗	✗
Penalty	✗	✗	✗	✗	✗	✗	✗	✗	✗
Super-Additivity	✗	✗	✗	✗	✗	✗	✗	✗	✗
MI-Separability	✗	✗	✗	✗	✗	✗	✗	✗	✗
Relative Separability	✗	✓	✗	✗	✓	✗	✗	✓	✗
Dominance	✗	✗	✓	✗	✗	✗	✗	✗	✗
Exchange	✓	✗	✓	✗	✗	✗	✗	✗	✗
Adjunction Invariance	✓	✓	✓	✗	✗	✗	✓	✓	✓

✓: satisfied, ✗: not satisfied.

Semantic inconsistency measures offer some useful tools for analysing inconsistent information (such as inconsistent requirements specifications in software engineering and conflicting reports in intelligence analysis). Furthermore, as illustrated by the above example, semantic inconsistency measures provide complementary insights compared to syntactic measures. The latter focus on the formulas, and consider which formulas are involved in inconsistencies, whereas the former focus on the atoms in the language, and consider which atoms are involved in inconsistencies.

### 8. Satisfaction of postulates for semantic inconsistency measures

In Table 2 we showed the satisfaction of some postulates for the inconsistency measures given earlier. In this section we write a similar table, Table 11, for the 9 inconsistency measures introduced in the previous section.

**Theorem 8.** *The satisfaction of the postulates for the semantic inconsistency measures introduced in the previous section is as given in Table 11. The symbol  $K$  is used throughout for different knowledgebases as explained for each item.*

**Proof.** We show the results individually for each inconsistency measure, by explaining why a postulate holds or giving a counterexample. As  $I_R^A$ , the semantic repair measure for A-logic, is identical to the contension measure,  $I_C$ , whose satisfaction of postulates, except for Relative Separability, is already known (see [52]), we deal only with a single postulate for it.

- $I_P^A$ 
  - (Monotony) The addition of a formula cannot reduce the number of problematic atoms.
  - (Free-Formula Independence) Let  $K = \{a \wedge \neg a \wedge b, \neg b\}$ . As mentioned earlier, this is an example of an iceberg inconsistency. The formula  $\neg b$  is free. But  $I_P^A(K) = 2 \neq I_P^A(K \setminus \{\neg b\}) = 1$ .
  - (Penalty) Let  $K = \{a \wedge b, \neg a \wedge \neg b, \neg a \wedge b\}$ . The formula  $\neg a \wedge b$  is problematic. Then  $I_P^A(K) = 2$  while  $I_P^A(K \setminus \{\neg a \wedge b\}) = 2$  also.
  - (Super-Additivity) Let  $K_1 = \{a \wedge b, \neg a \wedge \neg b\}$  and  $K_2 = \{\neg a \wedge b, a \wedge \neg b\}$ . Then  $K_1 \cap K_2 = \emptyset$ , but  $I_P^A(K_1 \cup K_2) = 2 \not\leq I_P^A(K_1) + I_P^A(K_2) = 4$ .
  - (MI-Separability) Let  $K_1 = \{a \wedge \neg a \wedge c \wedge d\}$  and  $K_2 = \{b \wedge \neg b \wedge c \wedge \neg c\}$ . Then the MI-Separability condition holds but  $I_P^A(K_1 \cup K_2) = 3 \neq I_P^A(K_1) + I_P^A(K_2) = 2$ .
  - (Relative Separability) Let  $K_1 = \{a \wedge \neg a\}$  and  $K_2 = \{b \wedge \neg b \wedge c \wedge \neg c\}$ . Then the Relative Separability condition holds but  $I_P^A(K_1) = 1 < I_P^A(K_2) = 2$  while  $I_P^A(K_1 \cup K_2) = 3$  which is not between 1 and 2.
  - (Dominance) Let  $K = \{\neg a, \neg b\}$ ,  $\phi = a$ ,  $\psi = a \vee b$ . The Dominance condition is satisfied but  $I_P^A(K \cup \{\phi\}) = 1 \not\leq I_P^A(K \cup \{\psi\}) = 2$ .
  - (Exchange) By Propositions 1 and 3 we may assume that  $K$ ,  $K'$ , and  $K''$  are all in CNF. The equivalence of  $K'$  and  $K''$  means that essentially they have the same CNF, meaning that the differences are trivial, such as the order of the clauses. But then the CNF for  $K \cup K'$  and  $K \cup K''$  are also essentially the same and hence the semantic inconsistency graphs for  $K \cup K'$  and  $K \cup K''$  must be the same.
  - (Adjunction Invariance) This follows from Proposition 3 as the single version of  $\{\phi, \psi\}$  is  $\phi \wedge \psi$ .
- $I_{P(r)}^A$ 
  - (Monotony) Let  $K_1 = \{a, \neg a\}$  and  $K_2 = K_1 \cup \{b\}$ . Then,  $K_1 \subset K_2$  but  $I_{P(r)}^A(K_1) = 1 \not\leq I_{P(r)}^A(K_2) = \frac{1}{2}$ .
  - (Free-Formula Independence) The same example can be used as for  $I_P^A$ .  $I_{P(r)}^A(K) = 1 \neq I_{P(r)}^A(K \setminus \{\neg b\}) = \frac{1}{2}$ .
  - (Penalty) The same example can be used as for  $I_P^A$ . Then  $I_{P(r)}^A(K) = 1$  while  $I_{P(r)}^A(K \setminus \{\neg a \wedge b\}) = 1$  also.
  - (Super-Additivity) The same example can be used as for  $I_P^A$ . Then  $I_{P(r)}^A(K_1 \cup K_2) = 1 \not\leq I_{P(r)}^A(K_1) + I_{P(r)}^A(K_2) = 2$ .
  - (MI-Separability) The same example can be used as for  $I_P^A$ . Then  $I_{P(r)}^A(K_1 \cup K_2) = \frac{3}{4} \neq I_{P(r)}^A(K_1) + I_{P(r)}^A(K_2) = \frac{5}{6}$ .
  - (Relative Separability) Let  $I_{P(r)}^A(K) = \frac{a}{b} < I_{P(r)}^A(K') = \frac{c}{d}$ . Using the condition about the atoms,  $I_{P(r)}^A(K \cup K') = \frac{a+c}{b+d}$  and indeed,  $\frac{a}{b} < \frac{a+c}{b+d} < \frac{c}{d}$ . The case for  $=$  is similar but simpler.
  - (Dominance) The same example can be used as for  $I_P^A$ . Then  $I_{P(r)}^A(K \cup \{\phi\}) = \frac{1}{2} \not\leq I_{P(r)}^A(K \cup \{\psi\}) = 1$ .
  - (Exchange) Let  $K = \{a\}$ ,  $K' = \{\neg a\}$ , and  $K'' = \{\neg a, b \vee \neg b\}$ . Then,  $K' \equiv K''$  but  $I_{P(r)}^A(K \cup K') = 1 \neq I_{P(r)}^A(K \cup K'') = \frac{1}{2}$ .
  - (Adjunction Invariance)  $K \cup \{\phi, \psi\}$  and  $K \cup \{\phi \wedge \psi\}$  both have the same number of problematic atoms and atoms.
- $I_R^A$ 
  - (Relative Separability) Let  $K_1 = \{a \wedge \neg a\}$  and  $K_2 = \{b \wedge \neg b\}$ . Then the Relative Separability condition holds but  $I_R^A(K_1) = I_R^A(K_2) = 1$  and  $I_R^A(K_1 \cup K_2) = 2$ .
- $I_P^B$ 
  - (Monotony) The addition of a formula cannot reduce the number of problematic atoms.
  - (Free-Formula Independence) Let  $K = \{\phi_1, \dots, \phi_m, \phi_{m+1}\}$  where  $\phi_{m+1}$  is free. If  $\text{Atoms}(\{\phi_1, \dots, \phi_m\} \cap \text{Atoms}(\{\phi_{m+1}\})) = \emptyset$  then the result is clear. Otherwise, let  $a \in \text{Atoms}(\{\phi_1, \dots, \phi_m\} \cap \text{Atoms}(\{\phi_{m+1}\}))$ . If  $a \in S$  where  $S \in \text{MAS}(K)$  then  $S \in \text{MAS}(K \setminus \{\phi_{m+1}\})$  because  $a$  must be a part of a formula and in B-logic it suffices to give any atom the value B to resolve the inconsistency in a formula. If  $a \notin S$  for any  $S \in \text{MAS}(K)$ , then  $a$  is not a problematic atom. In all cases  $I_P^B(K) = I_P^B(K \setminus \{\phi_{m+1}\})$ .
  - (Penalty) The same example can be used as for  $I_P^A$ .
  - (Super-Additivity) The same example can be used as for  $I_P^A$ .
  - (MI-Separability) The same example can be used as for  $I_P^A$ . Now  $I_P^B(K_1 \cup K_2) = 3 \neq I_P^B(K_1) + I_P^B(K_2) = 5$ .
  - (Relative Separability) The same example can be used as for  $I_P^A$ .
  - (Dominance) Let  $K = \{\neg b\}$ ,  $\phi = b$ , and  $\psi = b \wedge (a \vee \neg a)$ . Then the conditions for Dominance hold but  $I_P^B(K \cup \{\phi\}) = 1 \not\leq I_P^B(K \cup \{\psi\}) = 2$ .
  - (Exchange) The same example can be used as for Dominance except that now  $K' = \{\phi\}$  and  $K'' = \{\psi\}$ .
  - (Adjunction Invariance) Let  $K = \{\neg b\}$ ,  $\phi = b$ , and  $\psi = a \vee \neg a$ . Then,  $I_P^B(K \cup \{\phi, \psi\}) = 1 \neq I_P^B(K \cup \{\phi \wedge \psi\}) = 2$ .
- $I_{P(r)}^B$ 
  - (Monotony) The same example can be used as for  $I_{P(r)}^A$ .
  - (Free-Formula Independence) Let  $K = \{a \wedge \neg a, b\}$ . In  $K$  the formula  $b$  is free but  $I_{P(r)}^B(K) = \frac{1}{2} \neq I_{P(r)}^B(K \setminus \{b\}) = 1$ .
  - (Penalty) The same example can be used as for  $I_{P(r)}^A$ .
  - (Super-Additivity) The same example can be used as for  $I_{P(r)}^A$ .
  - (MI-Separability) The same example can be used as for  $I_{P(r)}^A$ . But now  $I_{P(r)}^B(K_1 \cup K_2) = 1 \neq I_{P(r)}^B(K_1) + I_{P(r)}^B(K_2) = 2$ .
  - (Relative Separability) The same reasoning can be used as for  $I_{P(r)}^A$ .
  - (Dominance) The same example can be used as for  $I_{P(r)}^A$ .

- (Exchange) The same example can be used as for  $I_{P(r)}^A$ .
- (Adjunction Invariance) The same example can be used as for  $I_P^B$ . But now  $I_P^B(K \cup \{\phi, \psi\}) = \frac{1}{2} \neq I_P^B(K \cup \{\phi \wedge \psi\}) = 1$ .
- $I_R^B$ 
  - (Monotony) The addition of a formula cannot reduce this value.
  - (Free Formula Independence) The same reasoning can be used as for  $I_P^B$ .
  - (Penalty) The same example can be used as for  $I_P^B$  but now  $I_R^B(K) = 1 = I_R^B(K \setminus \{-a \wedge b\})$ .
  - (Super Additivity) The same example can be used as for  $I_P^B$  but now  $I_R^B(K_1 \cup K_2) = 1 \not\geq I_R^B(K_1) + I_R^B(K_2) = 2$ .
  - (MI-Separability) The same example can be used as for  $I_P^B$  but now  $I_R^B(K_1 \cup K_2) = 1 \not\geq I_R^B(K_1) + I_R^B(K_2) = 2$ .
  - (Relative Separability) The same example can be used as for  $I_P^B$ , but now  $I_R^B(K_1) = I_R^B(K_2) = 1$ , while  $I_R^B(K_1 \cup K_2) = 2$ .
  - (Dominance) Let  $K = \{a, b \wedge \neg b\}$ ,  $\phi = \{-a\}$ , and  $\psi = \{-a \wedge (b \vee \neg b)\}$ . The condition for Dominance is satisfied, but  $I_R^B(K \cup \{\phi\}) = 2 \neq I_R^B(K \cup \{\psi\}) = 1$ .
  - (Exchange) The same example can be used as for Dominance with  $K' = \{\phi\}$  and  $K'' = \{\psi\}$ .
  - (Adjunction Invariance) Let  $K = \{a, b \wedge \neg b\}$ ,  $\phi = \neg a$ , and  $\psi = b \vee \neg b$ . Then  $I_R^B(K \cup \{\phi, \psi\}) = 2 \neq I_R^B(K \cup \{\phi \wedge \psi\}) = 1$ .
- $I_P^C$ 
  - (Monotony) The addition of a formula cannot reduce this value.
  - (Free Formula Independence) The same example can be used as for  $I_P^A$ .
  - (Penalty) The same example can be used as for  $I_P^A$ .
  - (Super-Additivity) The same example can be used as for  $I_P^A$ .
  - (MI-Separability) The same example can be used as for  $I_P^A$ .
  - (Relative Separability) The same example can be used as for  $I_P^A$ .
  - (Dominance) Let  $K = \{\neg a, \neg c\}$ ,  $\phi = a \vee (b \wedge c)$ , and  $\psi = (a \vee b) \wedge (a \vee c)$ . Then the Dominance condition is satisfied, but  $I_P^C(K \cup \{\phi\}) = 1 \not\geq I_P^C(K \cup \{\psi\}) = 2$ . For the former knowledgebase the problematic atom is  $c$ ; for the latter, both  $a$  and  $c$  are problematic.
  - (Exchange) Use the same example as for dominance except that  $K' = \{\phi\}$  and  $K'' = \{\psi\}$ . Then the Exchange condition is satisfied but  $I_P^C(K \cup K') = 1 \neq I_P^C(K \cup K'') = 2$ .
  - (Adjunction Invariance) The proof for  $I_P^A$  can be used.
- $I_{P(r)}^C$ 
  - (Monotony) The same example can be used as for  $I_{P(r)}^A$ .
  - (Free Formula Independence) The same example can be used as for  $I_{P(r)}^B$ .
  - (Penalty) The same example can be used as for  $I_{P(r)}^B$ .
  - (Super-Additivity) The same example can be used as for  $I_{P(r)}^A$ .
  - (MI-Separability) The same example can be used as for  $I_{P(r)}^A$ .
  - (Relative Separability) The same reasoning can be used as for  $I_{P(r)}^A$ .
  - (Dominance) The same example can be used as for  $I_{P(r)}^A$ .
  - (Exchange) The same example can be used as for  $I_{P(r)}^A$ .
  - (Adjunction Invariance) The proof for  $I_P^A$  can be used.
- $I_R^C$ 
  - (Monotony) The addition of a formula cannot reduce this value.
  - (Free Formula Independence) The same example can be used as for  $I_P^A$ .
  - (Penalty) The same example can be used as for  $I_P^A$ .
  - (Super-Additivity) The same example can be used as for  $I_P^A$ .
  - (MI-Separability) The same example can be used as for  $I_P^A$ .
  - (Relative Separability) The same example can be used as for  $I_P^A$ .
  - (Dominance) The same example can be used as for  $I_P^C$ .
  - (Exchange) The same example can be used as for  $I_P^C$ .
  - (Adjunction Invariance) The proof for  $I_P^A$  can be used. □

Considering Tables 2 and 11 we note first that Monotony is appropriate for absolute inconsistency measures and Relative Separability is appropriate for relative inconsistency measures; it does not matter if the measure is syntactic or semantic. But for the other postulates the syntactic inconsistency measures are more likely to satisfy the first group of postulates that involve relationships concerning sets of formulas while the semantic inconsistency measures are better at satisfying the last group that involves logical implication. The reason for this is that the first group of postulates can be thought of as syntactic postulates giving conditions about sets of formulas, while the second group deals with the semantics of the knowledgebase.

### 9. Additional postulates for semantic measures

As the semantic inconsistency measures were defined using the interpretation of the atoms, it is not surprising that they are not good at satisfying properties concerning sets of formulas. There is a similarity here with an approach to measuring inconsistency in spatio-temporal databases (see [23]) where the data has dimensions. For that reason new dimensional inconsistency measures were proposed that are more appropriate in that case. But the new inconsistency measures did not satisfy the standard postulates, so we proposed new postulates that took the dimensionality into account. In this section we define new postulates that are based on sets of atoms, like the semantic inconsistency measures, rather than on sets of formulas. In particular we rewrite the first four properties as semantic properties. This is similar to the way that we reformulated syntactic inconsistency bigraphs to semantic inconsistency bigraphs. As we can write each of these postulates in terms of the deletion of a set of formulas, we need to find the analogous operation for the semantic case. In a knowledgebase it is possible to delete formulas and so we need a counterpart for atoms. It turns out that the right choice is to substitute for the deletion of a set of formulas the modification of the interpretation of a set of atoms to give them the  $B$  value as we explain next.

In Section 5 we started with a classical 2-valued interpretation  $i$  and introduced the notation  $i[S]$  for the interpretation that assigns every atom in the set of atoms  $S$  the value  $B$ . In order to define the semantic postulates it is convenient to be able to start with an interpretation that is only partly classical: it allows three truth values for some atoms.

**Definition 11.** Let  $D$  be a set of atoms (in  $\text{Atoms}(K)$ ) and let  $i$  be a 2-valued interpretation. We write  $i_D$  for the interpretation that is 2-valued except that for the elements of  $D$  it is 3-valued.

$$i_D(a) = \begin{cases} B & \text{if } a \in D \\ i(a) & \text{otherwise} \end{cases}$$

Then the bracket notation is used as before. For sets of atoms  $D$  and  $S$ ,

$$i_D[S](a) = \begin{cases} B & \text{if } a \in S \\ i_D(a) & \text{otherwise} \end{cases}$$

In particular, if for some interpretation  $i_D, i_D[S] \models_L K$ , we say that  $S$  is an **L-atomic subset for  $K$  given  $D$** . An alternative notation will be useful here. Namely we write  $i[S] \models_L K_D$  and say that  $S$  is an **L-atomic subset for  $K_D$** . If there is no  $S' \subset S$  such that  $S'$  is an L-atomic subset for  $K_D$ , then  $S$  is a **minimal L-atomic subset for  $K_D$** . We use the notation  $\text{MAS}_L(K_D)$  to indicate the set of all minimal L-atomic subsets for  $K$  given  $D$ .

We use the notation  $K_D$  to reflect how the set of atoms  $D$  needs to be taken into account when an interpretation is considered for the knowledgebase  $K$ . So for instance, if we have  $K_{D_1}$  and  $K_{D_2}$ , then we have the same knowledgebase (i.e.  $K$ ) but if  $D_1 \neq D_2$ , then the interpretations do not necessarily treat the knowledgebases in the same way. Our motivation for this notation will become clearer later when we introduce atomic versions of rationality postulates in Definition 12.

Even though the role of the set  $D$  is different from the role of the set  $S$ , as we shall make clear below, there is an interplay between the sets of atoms  $D$  and  $S$  as shown by the following result.

**Proposition 8.** If  $D \cup S = D' \cup S'$ , then  $i_D[S] \models_L K$  iff  $i_{D'}[S'] \models_L K$ .

**Proof.** Assume  $D \cup S = D' \cup S'$ . Let  $i$  be a classical 2-valued interpretation. By Definition 11

$$i_{D \cup S}(a) = \begin{cases} B & \text{if } a \in D \cup S \\ i(a) & \text{otherwise} \end{cases}$$

Thus, if  $D \cup S = D' \cup S'$  then for all atoms  $a$ ,  $i_{D \cup S}(a) = i_{D' \cup S'}(a)$ . Hence, for every formula  $\phi$ ,  $i_{D \cup S}(\phi) = i_{D' \cup S'}(\phi)$ . So,  $i_D[S] \models_L \phi$  iff  $i_{D'}[S'] \models_L \phi$ , from which the result follows.  $\square$

We can extend the concept of semantic inconsistency bigraph to cover this more general case. That is, we can use the minimal L-atomic subsets for  $K_D$ . For illustration we redraw Figs. 4 - 6 to Figs. 12 - 14. We omit the atoms in  $D$  as they are no longer atoms whose truth values may have to be changed. To make things clearer we leave the space for the deleted atoms. Essentially, it is just a matter of removing the nodes and edges associated with the atoms in  $D$ . However, in the case of Fig. 12 we also delete an edge that no longer represents a minimal L-atomic subset.

The semantic inconsistency measures are defined as in Definition 10, except that now we use the semantic inconsistency bigraphs given for the appropriate logic and set. This allows for defining the semantic versions of the postulates involving sets of formulas in Definition 12. This involves changing a deletion of a formula to the deletion of an atom by using the given  $D$ . We previously introduced the concept of a problematic atom in Section 7. Now we formulate this concept and its opposite in terms of the semantic inconsistency graph. We write  $\text{FreeAtoms}(K_D)$  for the set of **free atoms** in  $K_D$ : these are the atoms whose corresponding  $u_i$  in the semantic inconsistency graph for  $K_D$  have  $\text{Deg}(u_i) = 0$ . We write  $\text{ProblematicAtoms}(K_D)$  for



Fig. 12. The semantic inconsistency bigraph for  $K_{[a,c]}$  where  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in A-logic.



Fig. 13. The semantic inconsistency bigraph for  $K_{[a,c]}$  where  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in B-logic.

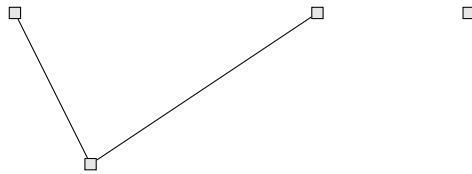


Fig. 14. The semantic inconsistency bigraph for  $K_{[a,c]}$  where  $K = \{a \wedge \neg a, b, \neg b, b \wedge \neg c, c \vee d, \neg d, c \vee e\}$  in C-logic.

Table 12  
Status of semantic postulates for semantic inconsistency measures.

Inconsistency measures	$I_P^L$	$I_{P(r)}^L$	$I_R^L$
Atomic Monotony	✓	✗	✓
Free-Atom Independence	✓	✗	✓
Atomic Penalty	✓	✓	✗
Atomic Super-Additivity	✓	✗	✗

✓: satisfied, ✗: not satisfied.

the set of **problematic atoms** whose corresponding  $u_i$  in the semantic inconsistency graph for  $K_D$  have  $\text{Deg}(u_i) > 0$ . In the next definition we provide the semantic counterpart to four postulates. Free-Atom Independence is the semantic counterpart to Free-Formula Independence.

**Definition 12.** The following are atomic versions of rationality postulates for semantic inconsistency measures.

**Atomic Monotony** If  $D' \subseteq D$  then  $I(K_D) \leq I(K_{D'})$ .

**Free-Atom Independence** If  $a \in \text{FreeAtoms}(K_D)$ , then  $I(K_D) = I(K_{D \cup \{a\}})$ .

**Atomic Penalty** If  $a \in \text{ProblematicAtoms}(K_D)$ , then  $I(K_D) > I(K_{D \cup \{a\}})$ .

**Atomic Super-Additivity** If  $\{D_1, D_2\}$  is a partition of  $\text{Atoms}(K)$  then  $I(K) \geq I(K_{D_1}) + I(K_{D_2})$ .

Our first result is that the identity of the logic for the satisfaction of these postulates is irrelevant.

**Proposition 9.** A semantic inconsistency measure either satisfies or violates a semantic postulate for all three logics.

**Proof.** These postulates refer to changes in the semantic inconsistency bigraph. Once this bigraph is created for the logic, the postulates refer only to specified changes in the bigraph. □

Using this proposition, when we consider the satisfaction of the semantic postulates we need to consider only three semantic inconsistency measures, namely  $I_P^L$ ,  $I_{P(r)}^L$ , and  $I_R^L$ . Our final result shows that the semantic inconsistency measures are more likely to satisfy the semantic versions of postulates than the standard syntactic versions.

**Theorem 9.** The satisfaction of the semantic postulates for the semantic inconsistency measures is as given in Table 12. That is,  $I_P^L$  satisfies Atomic Monotony, Free-Atom Independence, Atomic Penalty, and Atomic Super-Additivity;  $I_{P(r)}^L$  satisfies Atomic Penalty; and  $I_R^L$  satisfies Atomic Monotony and Free-Atom Independence.



**Proof.** It will be convenient to do the proof separately for each semantic postulate.

- (Atomic Monotony) In the case of  $D' \subseteq D$ , the inconsistency bigraph for  $K_D$  is obtained from the inconsistency bigraph for  $K_{D'}$  by possibly deleting some of the elements of  $U$  as well as the corresponding edges and possibly elements of  $V$ . While the elements deleted need not correspond to problematic atoms,  $I_p^L(K_D) \leq I_p^L(K_{D'})$ . Also, as elements are deleted from  $U$ ,  $\min\{\text{Deg}(v) \mid v \in V\}$  cannot increase. Hence  $I_R^L(K_D) \leq I_R^L(K_{D'})$ . But consider a case such as the one in Fig. 6. Let  $D = \{e\}$  and  $D' = \emptyset$ . Then  $I_{P(r)}^L(K_D) = 1 \not\leq I_{P(r)}^L(K_{D'}) = \frac{4}{5}$ .
- (Free-Atom Independence) The deletion of an element  $u \in U$  for which  $\text{Deg}(u) = 0$  does not change  $I_p^L$  (the number of problematic atoms) or  $I_R^L$  (the minimal degree of an element of  $V$ ). However, it does change  $I_{P(r)}^L$ , as shown in the example for Atomic Monotony, because  $|U|$  decreases by 1 but  $|V|$  stays the same.
- (Atomic Penalty) In this case the element  $u \in U$  that is deleted corresponds to a problematic atom. This reduces the number of problematic elements. Hence  $I_p^L$  satisfies this postulate. So does  $I_{P(r)}^L$  because the total number of elements deleted is at least 1; hence the fraction is reduced. The situation is different for  $I_R^L$ . For example, in the case of Fig. 7, if  $a$  is deleted,  $I_R^L$  does not change.
- (Atomic Super-Additivity) As  $D_1 \cap D_2 = \emptyset$ , the sum of the number of problematic elements in the two portions is at most the number of problematic elements in the inconsistency bigraph for  $K$ . Consider now the same example as for Atomic Monotony where  $D_1 = \{a, b, c, d\}$  and  $D_2 = \{e\}$ . Then the condition is satisfied but  $I_{P(r)}^L(K) = \frac{4}{5} \not\leq I_{P(r)}^L(K_{D_1}) + I_{P(r)}^L(K_{D_2}) = 1$ . For  $I_R^L$  consider Fig. 7 with  $D_1 = \{a, b\}$  and  $D_2 = \{c, d, e\}$ . Again the condition is satisfied, but  $I_R^L(K) = 1 \not\leq I_R^L(K_{D_1}) + I_R^L(K_{D_2}) = 2$ .  $\square$

In this section, we have introduced four new postulates. Each of them is intended to capture how semantic modifications to the knowledgebase affect the inconsistency measures. Using these new postulates we can distinguish between the three types of semantic inconsistency measures, and furthermore, one type of the measures, namely  $I_p^L$ , satisfies all four postulates.

### 10. Summary and future work

In this paper, we have provided a novel framework for comparing semantic inconsistency measures based on 3-valued logics. Existing proposals for inconsistency measures fit into this framework, and we have used the framework to identify novel inconsistency measures. The framework is based on the notion of inconsistency bigraphs and is analogous to inconsistency bigraphs for syntactic measures. However, in contrast to the syntactic measures, which tend to be based on classical logic, here we have considered three different 3-valued logics. So we see the framework as being applicable to a variety of many-valued logics. As part of the analysis the novel inconsistency measures, we have used existing postulates for inconsistency measures. We have also introduced new postulates that have been developed to give insights into semantic inconsistency measures. These postulates allow us to compare and contrast different inconsistency measures.

As we have argued in the introduction, an atom-centric approach to inconsistency measures, as exemplified by semantic inconsistency measures, has some advantages over a formula-centric approach to inconsistency measures. Yet there is much more investigation of formula-centric approaches in the literature. So this paper is a contribution to filling in the gap in our understanding of atom-centric approaches.

Through the investigations in this paper, we have a better understanding of the underlying nature of measuring inconsistency based on 3-valued logics, and we have some novel alternatives for inconsistency measures. Furthermore, we have some insights into how the choice of the 3-valued logic affects the inconsistency measure. This investigation may therefore facilitate the uptake of semantic inconsistency measures in applications where there is a need to avoid syntax sensitivity, and where syntactic approaches lack discrimination between those atoms that are potentially responsible for an inconsistency, and those that are “uninvolved” in a conflict.

So the semantic measures considered in this paper provide a valuable alternative to the better-known syntactic measures. We have shown via analysis of the postulates, that they differ in various ways. Nonetheless, further comparison with syntactic approaches would be valuable including with those based on generalized consequence relations [15].

Given that our framework allows different logics to be used as the basis of inconsistency measures, it would be desirable to consider relations between types of logic, and properties of the inconsistency measures that are based on them. For instance, the following is a relation between logics.

$L$  is **more relaxed** than  $L'$  iff for all  $\phi$ , and for all interpretations  $i$ , if  $i \models_L \phi$ , then  $i \models_{L'} \phi$

Then using this relation, we could consider a property of inconsistency measures such as the following which assumes that more relaxed logics lead to lower values for inconsistency measures.

If  $L$  is more relaxed than  $L'$ , then for all  $K$ ,  $I^L(K) \leq I^{L'}(K)$

**Table 13**

A four-valued logic by Belnap [3] where  $B$  denotes that truth value is both  $T$  and  $F$ , and  $N$  denotes that truth value is neither  $T$  or  $F$ . For this,  $T$  is ranked higher than  $B$  and  $N$  and  $N$  and  $B$  are ranked higher than  $F$  (i.e.  $T > B$ ,  $T > N$ ,  $B > F$ , and  $N > F$ ). Below are the truth tables for negation (left), conjunction (centre), and disjunction (right).

$\alpha$	$N$	$F$	$T$	$B$
$\neg\alpha$	$N$	$T$	$F$	$B$

$\wedge$	$N$	$F$	$T$	$B$
$N$	$N$	$F$	$N$	$F$
$F$	$F$	$F$	$F$	$F$
$T$	$N$	$F$	$T$	$B$
$B$	$F$	$F$	$B$	$B$

$\vee$	$N$	$F$	$T$	$B$
$N$	$N$	$N$	$T$	$T$
$F$	$N$	$F$	$T$	$B$
$T$	$T$	$T$	$T$	$T$
$B$	$T$	$B$	$T$	$B$

However, this is not a property that holds for the 3-valued logics and inconsistency measures considered in this paper. Nonetheless, it will be worthwhile in future work to consider properties of inconsistency measures based on the underlying logic.

In future work, we will also investigate further possibilities for inconsistency measures based on the structure of the semantic inconsistency bigraph  $(U, V, E)$  for a knowledgebase  $K$ . These include counting the number of edges (i.e.  $I(K) = |E|$ ) or counting the number of minimal L-atomic subsets (i.e.  $I(K) = |V|$ ). These measure the “complexity” of the inconsistencies in a knowledgebase, and reflect the work that is required for understanding the inconsistencies, perhaps prior to deciding on resolution actions.

Finally, in future work, we will consider other 3-valued logics to see what happens if some of the six properties of Theorem 2 are relaxed and extend the approach to 4-valued and many valued logics. For instance, Belnap’s logic (see Fig. 13) and QC logic [29] are 4-valued logics that can be used as the basis of inconsistency measures using the notion of contension [30,20]. By extending the proposals in this paper to incorporate 4-valued and many-valued logics, we will have a richer range of options for analysing inconsistency. There are advantages of 4-valued logic over 3-valued logic for reasoning with inconsistent information (see for example [1]). We will also consider connections with previous work on measuring inconsistency in many valued logics (see [51]).

**Declaration of competing interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Data availability**

No data was used for the research described in the article.

**Acknowledgement**

We wish to thank Davide Ciucci for giving us information about the three papers concerning C-logic. We are also grateful to the reviewers for many very helpful comments and suggestions.

**References**

- [1] O. Arieli, A. Avron, The value of the four values, *Artif. Intell.* 102 (1998) 97–141.
- [2] A. Avron, Natural 3-valued logics—characterization and proof theory, *J. Symb. Log.* 56 (1) (1991) 276–294.
- [3] N. Belnap, A useful four-valued logic, in: *Modern Uses of Multiple-Valued Logic*, Reidel, 1977, pp. 8–37.
- [4] L. Bertossi, Measuring and computing database inconsistency via repairs, Technical Report, arXiv:1804.08834, 2018.
- [5] L. Bertossi, Repair-based degrees of database inconsistency, in: *Logic Programming and Nonmonotonic Reasoning (Proceedings of LPNMR’19)*, in: *Lecture Notes in Computer Science*, vol. 11481, Springer, 2019, pp. 195–209.
- [6] Ph. Besnard, Revisiting postulates for inconsistency measures, in: *Logics in Artificial Intelligence (JELIA’14)*, in: LNCS, Springer, 2014, pp. 383–396.
- [7] Ph. Besnard, J. Grant, Relative inconsistency measures, *Artif. Intell.* 280 (2020) 103231.
- [8] Ph. Besnard, A. Hunter, Quasi-classical logic: non-trivializable classical reasoning from inconsistent information, in: *Proceedings of ECSQARU’95*, in: *Lecture Notes in Computer Science*, vol. 946, Springer, 1995, pp. 44–51.
- [9] L. Bertossi, A. Hunter, T. Schaub, Introduction to inconsistency tolerance, in: *Inconsistency Tolerance*, in: LNCS, vol. 3300, Springer, 2005, pp. 1–14.
- [10] D. Ciucci, D. Dubois, Three-valued logics, uncertainty management and rough sets, *Trans. Rough Sets* 17 (2014) 1–32.
- [11] C. Corea, J. Grant, M. Thimm, Measuring inconsistency in declarative process specifications, in: *Business Process Management - Proceedings of BPM’22*, in: *Lecture Notes in Computer Science*, vol. 13420, Springer, 2022, pp. 289–306.
- [12] J.-F. Condotta, B. Raddaoui, Y. Salhi, Quantifying conflicts for spatial and temporal information, in: *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR’16)*, IEEE Press, 2016, pp. 443–452.
- [13] G. De Bona, M. Finger, Measuring inconsistency in probabilistic logic: rationality postulates and Dutch book interpretation, *Artif. Intell.* 227 (2015) 140–164.
- [14] G. De Bona, J. Grant, A. Hunter, S. Konieczny, Classifying inconsistency measures using graphs, *J. Artif. Intell. Res.* 66 (2019) 937–987.
- [15] G. De Bona, A. Hunter, Localizing iceberg inconsistencies, *Artif. Intell.* 246 (2017) 118–151.
- [16] H. Decker, D. Martinenghi, Inconsistency-tolerant integrity checking, *IEEE Trans. Knowl. Data Eng.* 23 (2) (2011) 218–234.
- [17] D. Dubois, H. Prade, Conditional objects as nonmonotonic consequence relationships, *IEEE Trans. Syst. Man Cybern.* 24 (1994) 1724–1740.

- [18] L. Fan, M. Flood, J. Grant, Measuring inconsistency in bank holding data, in: Proceedings of the Fifth International Workshop on Data Science for Macro-Modeling (DSMM'19), ACM, 2019, pp. 1–5.
- [19] D. Gabbay, A. Hunter, Making inconsistency respectable 1: a logical framework for inconsistency in reasoning, in: Foundations of Artificial Intelligence Research, vol. 535, Springer, 1991, pp. 19–32.
- [20] J. Grant, A. Hunter, Measuring inconsistency in knowledgebases, *J. Intell. Inf. Syst.* 27 (2006) 159–184.
- [21] J. Grant, A. Hunter, Analysing inconsistent first-order knowledge bases, *Artif. Intell.* 172 (2008) 1064–1093.
- [22] J. Grant, A. Hunter, Measuring the good and bad in inconsistent information, in: Proceedings of the International Joint Conference on AI (IJCAI'11), 2011, pp. 2632–2637.
- [23] J. Grant, M.V. Martinez, C. Molinaro, F. Parisi, Dimensional inconsistency measures and postulates in spatio-temporal databases, *J. Artif. Intell. Res.* 71 (2021) 733–780.
- [24] J. Grant, Classifications for inconsistent theories, *Notre Dame J. Form. Log.* 19 (1978) 435–444.
- [25] J. Grant, Measuring inconsistency in some branching time logics, *J. Appl. Non-Class. Log.* 31 (2) (2021) 85–107.
- [26] A. Hunter, S. Konieczny, Approaches to measuring inconsistent information, in: Inconsistency Tolerance, in: LNCS, vol. 3300, Springer, 2005, pp. 189–234.
- [27] A. Hunter, S. Konieczny, Shapley inconsistency values, in: Proceedings of the 10th International Conference on Knowledge Representation (KR'06), AAAI Press, 2006, pp. 249–259.
- [28] A. Hunter, S. Konieczny, On the measure of conflicts: Shapley inconsistency values, *Artif. Intell.* 174 (2010) 1007–1026.
- [29] A. Hunter, Reasoning with conflicting information using quasi-classical logic, *J. Log. Comput.* 10 (2000) 677–703.
- [30] A. Hunter, Measuring inconsistency in knowledge via quasi-classical models, in: Proceedings of the National Conference on Artificial Intelligence (AAAI'02), MIT Press, 2002, pp. 68–73.
- [31] A. Hunter, Evaluating significance of inconsistencies, in: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03), 2003, pp. 468–473.
- [32] S. Konieczny, J. Lang, P. Marquis, Quantifying information and contradiction in propositional logic through epistemic actions, in: Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03), 2003, pp. 106–111.
- [33] K. Knight, Measuring inconsistency, *J. Philos. Log.* 31 (2002) 77–98.
- [34] E. Livshits, B. Kimelfeld, The Shapley value of inconsistency measures for functional dependencies, *Log. Methods Comput. Sci.* 18 (2) (2022).
- [35] E. Livshits, R. Kochirgan, S. Tsur, I. Ilyas, B. Kimelfeld, S. Roy, Properties of inconsistency measures for databases, in: Proceedings of SIGMOD'21, ACM, 2021, pp. 1182–1194.
- [36] K. Mu, Z. Jin, W. Liu, D. Zowghi, Bo Wei, Measuring the significance of inconsistency in the viewpoints framework, *Sci. Comput. Program.* 78 (2013) 1572–1599.
- [37] K. Mu, W. Liu, Z. Jin, D. Bell, A syntax-based approach to measuring the degree of inconsistency for belief bases, *Int. J. Approx. Reason.* 52 (2011) 978–999.
- [38] K. McAreevey, W. Liu, P. Miller, K. Mu, Measuring inconsistency in a network intrusion detection rule set based on snort, *Int. J. Semant. Comput.* 5 (3) (2011) 281–322.
- [39] N. Madrid, M. Ojeda-Aciego, Measuring inconsistency in fuzzy answer set semantics, *IEEE Trans. Fuzzy Syst.* 19 (4) (2011) 605–622.
- [40] Y. Ma, G. Qi, P. Hitzler, Z. Lin, Measuring inconsistency for description logics based on paraconsistent semantics, in: Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU'07), in: LNCS, vol. 4724, Springer, 2007, pp. 30–41.
- [41] K. Mu, Formulas free from inconsistency: an atom-centric characterization in Priest's minimally inconsistent LP, *J. Artif. Intell. Res.* 66 (2019) 279–296.
- [42] K. Mu, K. Wang, L. Wen, Preferential multi-context systems, *Int. J. Approx. Reason.* 75 (2016) 39–56.
- [43] F. Parisi, J. Grant, On measuring inconsistency in relational databases with denial constraints, in: Proceedings of ECAI'20, in: Frontiers in Artificial Intelligence and Applications, vol. 325, IOS Press, 2020, pp. 857–864.
- [44] N. Potyka, Linear programs for measuring inconsistency in probabilistic logics, in: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR'14), 2014, pp. 568–577.
- [45] G. Priest, The logic of paradox, *J. Philos. Log.* 8 (1) (1979) 219–241.
- [46] N. Potyka, M. Thimm, Probabilistic reasoning with inconsistent beliefs using inconsistency measures, in: Proceedings of IJCAI'15, AAAI Press, 2015, pp. 3156–3163.
- [47] G. Qi, A. Hunter, Measuring incoherence in description logic-based ontologies, in: The Semantic Web, 6th International Semantic Web Conference (ISWC'07), in: LNCS, vol. 4825, Springer, 2007, pp. 381–394.
- [48] N. Rescher, *Many-Valued Logic*, McGraw-Hill Inc, 1969.
- [49] B. Sobociński, Axiomatization of a partial system of three-value calculus of propositions, *J. Comput. Syst.* 1 (1952) 23–55.
- [50] M. Thimm, On the expressivity of inconsistency measures, *Artif. Intell.* 234 (2016) 120–151.
- [51] M. Thimm, Measuring inconsistency with many-valued logics, *Int. J. Approx. Reason.* 86 (2017) 1–23.
- [52] M. Thimm, On the evaluation of inconsistency measures, in: J. Grant, M.V. Martinez (Eds.), *Measuring Inconsistency in Information*, College Publications, 2018, pp. 169–194.
- [53] M. Ulbricht, M. Thimm, G. Brewka, Measuring inconsistency in answer set programs, in: Proceedings of JELIA'16, 2016, pp. 577–583.
- [54] E.A. Walker, Stone algebras, conditional events, and three-valued logic, *IEEE Trans. Syst. Man Cybern.* 24 (1994) 1699–1707.
- [55] L. Zhou, H. Huang, G. Qi, Y. Ma, Z. Huang, Yi. Qu, Measuring inconsistency in dl-lite ontologies, in: IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, 2009, IEEE, 2009, pp. 349–356.
- [56] X. Zhu, Z. Jin, Inconsistency measurement of software requirements specifications: an ontology-based approach, in: IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'05), IEEE Press, 2005, pp. 402–410.