International Journal of Approximate Reasoning 55 (2014) 529–556

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Approaches to measuring inconsistency for stratified knowledge bases ☆

Kedian Mu [a,*], Kewen Wang [b], Lian Wen [b]

[a] *School of Mathematical Sciences, Peking University, Beijing 100871, PR China*
[b] *School of Information and Communication Technology, Nathan Campus, Griffith University, 170 Kessels Road, Nathan, Brisbane, Queensland 4111, Australia*

ARTICLE INFO

Article history:
Received 25 June 2013
Received in revised form 9 October 2013
Accepted 20 November 2013
Available online 4 December 2013

Keywords:
Inconsistency measure
Stratified knowledge base

ABSTRACT

A number of proposals have been proposed for measuring inconsistency for knowledge bases. However, it is rarely investigated how to incorporate preference information into inconsistency measures. This paper presents two approaches to measuring inconsistency for stratified knowledge bases. The first approach, termed the multi-section inconsistency measure (MSIM for short), provides a framework for characterizing the inconsistency at each stratum of a stratified knowledge base. Two instances of MSIM are defined: the naive MSIM and the stratum-centric MSIM. The second approach, termed the preference-based approach, aims to articulate the inconsistency in a stratified knowledge base from a global perspective. This approach allows us to define measures by taking into account the number of formulas involved in inconsistencies as well as the preference levels of these formulas. A set of desirable properties are introduced for inconsistency measures of stratified knowledge bases and studied with respect to the inconsistency measures introduced in the paper. Computational complexity results for these measures are presented. In addition, a simple but explanatory example is given to illustrate the application of the proposed approaches to requirements engineering.

1. Introduction

Inconsistency is a pervasive and important problem in many applications when information is gathered from multiple sources. It has been increasingly recognized that measuring inconsistency plays an important role in analyzing inconsistent information in a variety of applications such as belief change [1], knowledge bases merging [2], ontology management [3], requirements engineering [4–6], expert systems in medicine [7] and intrusion detections in security [8]. A growing number of inconsistency measures for knowledge bases have been proposed recently [1,9–22].

Most of these inconsistency measures focus on classical knowledge bases only. Such measures assume that any two pieces of knowledge in the same knowledge base are equally preferred. However, some pieces of knowledge are often more preferred than others in some applications, such as belief revision [23], knowledge bases merging [24], and inconsistency management in requirements engineering [4,25]. More importantly, the preference relation between pieces of knowledge is always considered as useful to resolving inconsistency in these applications. Therefore, proposals for measuring inconsistency in such applications should take into account preferences on knowledge. To illustrate this, consider a scenario about selecting the best paper from submissions for a conference. Suppose that A and B are two conference submissions. Alice and Bob are the two reviewers for these two submissions. Alice recommended A for the best paper with *high confidence*, whilst

☆ Part of this work was done when the first author visited Griffith University.
* Corresponding author.

0888-613X/$ – see front matter
http://dx.doi.org/10.1016/j.ijar.2013.11.006

Bob gave the contrary reviewing result with *low confidence*. Bob recommended B for the best paper with *high confidence*, but Alice opposed Bob's recommendation with *high confidence*. Intuitively, the contradiction between Alice and Bob about A is different from that about B. Evidently, if we ignore recommendations with *low confidence*, B receives two valid reviews that are contradictory, whilst A has just one review that is positive. Further, if we consider all reviews, A also has two contradictory reviews. But the contradiction between Alice and Bob about A is less sharp than that about B, because Bob opposes Alice w.r.t. A with *low confidence* instead of *high confidence*.

It should be pointed out that while there is a lot of work on utilizing preference information for handling inconsistent knowledge bases (e.g. [26,27]), there are relatively few approaches for measuring inconsistency for knowledge bases using preference.

On the other hand, given a preference relation over a knowledge base, the base may be stratified into several strata according to the preference relation between pieces of knowledge, if we assume that any two pieces of knowledge in the same stratum are equally preferred. Moreover, each stratum of a given knowledge base, except for the least preferred stratum, naturally splits the whole knowledge base into two parts: one is the set of formulas more or equally preferred than the stratum, and the other is the set of formulas strictly less preferred than the stratum. In many applications, we need to consider only the first part of the knowledge base instead of the whole base. For example, to manage time effectively and efficiently, an agenda robot needs to classify a set of tasks into four strata, including the *urgent and important*, the *urgent and not important*, the *not urgent but important*, and the *not urgent and not important*. Some of these tasks may contradict to each other. Suppose an agenda robot has detected that the user's available time is very limited. Then it is advisable for the user to focus on only *urgent* tasks, i.e., the first one or the first two strata. Roughly speaking, compared to flat knowledge base, the stratum-based structure of a knowledge base makes it possible for the robot to focus on relevant parts in the stratified knowledge base. As a result, we need to measure the inconsistency of a stratified knowledge base stratum by stratum.

To address these issues, we propose two approaches to measuring inconsistency for stratified knowledge bases in this paper. The first approach, termed the multi-section inconsistency measure (MSIM for short), provides a framework for capturing the inconsistency occurring at each stratum of a stratified knowledge base. Informally, the multi-section inconsistency measure for a stratified knowledge base is a vector such that the first i components of the vector together exactly capture the inconsistency in the first i strata of the base. In particular, we present two instances in this framework, i.e., the naive MSIM and the stratum-centric MSIM. The former takes advantage of some intuitive inconsistency measures for flat knowledge bases such as the $\mathsf{LP_m}$ inconsistency measure presented in [1], to characterize inconsistencies at each stratum of a stratified knowledge base. The latter aims to capture the most preferred stratum involved in inconsistencies caused by each stratum of a stratified knowledge base, in which inconsistencies are characterized in terms of minimal inconsistent subsets of each i-cut (the union of the first i strata) of the base. The second approach to measuring inconsistency, termed preference-based approach, focuses on assessing inconsistencies in a whole stratified knowledge base from an integrated perspective. It allows us to define measures by considering the number of formulas involved in inconsistencies as well as the preference levels of these formulas under $\mathsf{LP_m}$ models, one of representative paraconsistent models [28]. By examining their logical properties, we show that these inconsistency measures indeed take into account the preorder relation on a stratified knowledge base in assessing the inconsistency in the base. Then we provide a systemic analysis of the computational complexity of these measures. Finally, we use a small but explanatory example to illustrate how to use our measures to monitor the process of inconsistency handling in requirements engineering.

The rest of this paper is organized as follows. Section 2 provides some necessary notations about inconsistency as well as stratified knowledge bases. In Section 3 we propose the multi-section inconsistency measure and its two instances for stratified knowledge bases. In Section 4 we propose preference-based inconsistency measures for stratified knowledge bases. In Section 5 we address logical properties of these inconsistency measures. In Section 6 we study computational complexity issues about these measures. In Section 7, we present an example to illustrate the application of our approaches in the domain of requirements engineering. In Section 8 we will compare our approaches with some closely related work. Finally, we conclude this paper in Section 9.

2. Preliminaries

In this section we introduce some basics about knowledge bases and inconsistency measures that will be used in the paper. We consider the language $\mathcal{L}$ built from a finite set of propositional variables $\mathcal{P}$ under logical connectives $\{\neg, \wedge, \vee, \rightarrow\}$ and constants $\mathbb{T}$ (true) and $\mathbb{F}$ (false). We use $a, b, c, \ldots$ to denote propositional variables, and $\alpha, \beta, \gamma, \ldots$ to denote formulas.

2.1. Stratified knowledge bases

Before introducing stratified knowledge bases, we first fix some necessary mathematical notations. By $(c_1, c_2, \ldots, c_n)$ or $\vec{\mathrm{c}}$ for short, we denote an n-ary vector. In particular, we use $\vec{0}$ to denote the zero vector $(0, 0, \ldots, 0)$. The lexicographical ordering relation $\leqslant$ between two vectors $\vec{\mathrm{c}} = (c_1, c_2, \ldots, c_n)$ and $\vec{\mathrm{d}} = (d_1, d_2, \ldots, d_n)$ of the same size is given as follows: $\vec{\mathrm{c}} \leqslant \vec{\mathrm{d}}$ if $\vec{\mathrm{c}} = \vec{\mathrm{d}}$, or there exists $k \leqslant n$ s.t. $c_k < d_k$ and $c_i = d_i$ for each $i < k$.

Furthermore, $\vec{\mathrm{c}} < \vec{\mathrm{d}}$ if and only if $\vec{\mathrm{c}} \leqslant \vec{\mathrm{d}}$ and $\vec{\mathrm{c}} \neq \vec{\mathrm{d}}$.

If $(c_1, c_2, \dots, c_n)$ is a vector and c_{n+1} is a real number, then the concatenation operation between $\vec{c}$ and c_{n+1}, denoted $(c_1, c_2, \dots, c_n) \uplus c_{n+1}$, is defined as

$$(c_1, c_2, \dots, c_n) \uplus c_{n+1} = (c_1, c_2, \dots, c_n, c_{n+1}).$$

A binary relation $\leqslant_A$ on some set A is a total preorder relation if it is reflexive, transitive, and total, i.e., for all $a, b, c \in A$, we have that:

(1) $a \leqslant_A a$ (reflexivity),
(2) if $a \leqslant_A b$ and $b \leqslant_A c$, then $a \leqslant_A c$ (transitivity),
(3) $a \leqslant_A b$ or $b \leqslant_A a$ (totality).

A *stratified knowledge base* (stratified KB for short), also termed *ranked knowledge base*, is a pair $K = (C_K, \leqslant_K)$, where C_K is a finite set of propositional formulas and $\leqslant_K$ is a total preorder on C_K [29]. For any $\alpha, \beta \in C_K$, $\alpha \leqslant_K \beta$ means that α is preferred to β in K.

A stratified knowledge base $K = (C_K, \leqslant_K)$ is often given in the form of a stratification $(S_1, S_2, \dots, S_n)$ for K such that

- $\emptyset \subset S_i \subseteq C_K$ for each $1 \leqslant i \leqslant n$;
- $\bigcup_{i=1}^{n} S_i = C_K$ and $S_i \cap S_j = \emptyset$ for $i \neq j$.
- For $\alpha, \beta \in C_K$, $\alpha \leqslant_K \beta$ iff $\alpha \in S_i$, $\beta \in S_j$ and $i \leqslant j$.

Each S_i is called the i-th *stratum* of the stratification. Throughout this paper, we use $K = (S_1, S_2, \dots, S_n)$ or just $(S_1, S_2, \dots, S_n)$ to denote the stratified knowledge base $K = (C_K, \leqslant_K)$ with n strata.

Given a stratified knowledge base $K = (S_1, S_2, \dots, S_n)$, we call $K(i) = \bigcup_{k=1}^{i} S_k$ the i-cut of K for each $1 \leqslant i \leqslant n$ [24]. Obviously, the n-cut of K is exactly K. We use $\mathrm{lc}(K)$ to denote the lexicographical cardinality of K, i.e., $\mathrm{lc}(K) = (|S_1|, |S_2|, \dots, |S_n|)$.

Furthermore, for each $1 \leqslant i \leqslant n$, we call the stratified knowledge base $K_{1\to i} = (S_1, S_2, \dots, S_i)$ the i-section of K. Roughly speaking, the i-cut of K is exactly a set consisting of formulas in the first i strata, whilst the i-section conveys the preorder relation among these formulas as well as these formulas.

A stratified knowledge base $K = (C_K, \leqslant_K)$ is inconsistent if C_K is inconsistent, i.e., there is a formula α such that $C_K \vdash \alpha$ and $C_K \vdash \neg\alpha$, where $\vdash$ is the classical inference relation. If there is no confusion, we use $K \vdash \bot$ to denote that K is inconsistent. For example, $K = (\{a\}, \{\neg a, b\})$ is inconsistent because $\{a, \neg a, b\}$ is inconsistent.

The total preorder relation over a stratified knowledge base is conveyed by the n-strata structure of the base. However, such a structure is not commensurable in many cases, and then it is difficult to compare any two stratified knowledge bases with different strata. To bear this in mind, we focus on only changes that cannot affect the n-strata structure of a stratified knowledge base. That is, we assume the scale of priority levels in a knowledge base always remains unchanged. Given a stratified knowledge base $K = (S_1, S_2, \dots, S_n)$, a propositional formula α and an integer i $(1 \leqslant i \leqslant n)$, we use $K \cup_i \{\alpha\}$ to denote the stratified knowledge base obtained by adding α to the i-th stratum of K, i.e., $K \cup_i \{\alpha\} = (S_1, S_2, \dots, S_i \cup \{\alpha\}, \dots, S_n)$. Note that such an enlargement of K does not change the n-strata structure of K.

On the other hand, in order to keep the n-strata structure within a stratified knowledge base, we also allow empty strata in the rest of paper. We call a stratified KB $K' = (S'_1, S'_2, \dots, S'_n)$ a sub-KB of another stratified KB $K = (S_1, S_2, \dots, S_n)$, denoted $K' \sqsubseteq K$, if $\emptyset \subseteq S'_i \subseteq S_i$ for all $1 \leqslant i \leqslant n$. In particular, for any $K', K'' \sqsubseteq K$, we denote $K' \preceq K''$ if $\mathrm{lc}(K') \leqslant \mathrm{lc}(K'')$. We use $\mathsf{LMI}(K)$ to denote the set of all minimal inconsistent subsets of K w.r.t. $\preceq$, that is,

$$\mathsf{LMI}(K) = \left\{K' \sqsubseteq K \;\middle|\; K' \vdash \bot,\ \forall K'' \sqsubset K' \text{ and } K' \prec K'',\ K'' \nvdash \bot\right\}.$$

Here L in LMI means lexicographical ordering, which is used to define $\mathrm{lc}(K)$.

2.2. Inconsistency measures for flat knowledge bases

If there is only one stratum in a stratified KB K, i.e., $K = (S_1)$, we call K a *flat* or classical knowledge base. In this case, we just write $K = S_1$ for simplicity. Obviously, any two formulas in a flat knowledge base are considered equally preferred. For convenience, we assume that a flat knowledge base in this paper is always not empty. In this subsection we recall some major inconsistency measures for flat knowledge bases.

To deal with inconsistent flat KBs, some paraconsistent semantics have been introduced. A well known approach to paraconsistency is based LP_m models used in Priest's Logic of Paradox (LP for short) [28]. An LP_m model is a three-valued model with truth values $\{\mathsf{T}, \mathsf{F}, \mathsf{B}\}$, in which the third truth value B means that a statement is both true and false in the model.

The following material about the LP_m model is largely taken from [13,22]. An $\mathrm{LP_m}$ interpretation (or just interpretation) ω in the Logic of Paradox maps each propositional formulas to one of the three truth values T, F, B such that

- $\omega(\mathbb{T}) = \mathsf{T}$, $\omega(\mathbb{F}) = \mathsf{F}$,
- $\omega(\neg\alpha) = \mathsf{B}$ if and only if $\omega(\alpha) = \mathsf{B}$,
- $\omega(\neg\alpha) = \mathsf{T}$ if and only if $\omega(\alpha) = \mathsf{F}$,
- $\omega(\alpha \wedge \beta) = \min_{\leqslant t}\{\omega(\alpha), \omega(\beta)\}$,
- $\omega(\alpha \vee \beta) = \max_{\leqslant t}\{\omega(\alpha), \omega(\beta)\}$,
- $\omega(\alpha \to \beta) = \begin{cases} \mathsf{T}, & \text{if } \omega(\alpha) = \mathsf{F} \\ \omega(\beta), & \text{otherwise,} \end{cases}$

where $\mathsf{F} <_t \mathsf{B} <_t \mathsf{T}$. Then the set of $\mathsf{LP_m}$ models of a propositional formula α is defined as

$$\mathsf{Mod_{LP}}(\alpha) = \{\omega \mid \omega(\alpha) \in \{\mathsf{T}, \mathsf{B}\}\}.$$

Further, the set of $\mathsf{LP_m}$ models of K is defined as

$$\mathsf{Mod_{LP}}(K) = \bigcap_{\alpha \in K} \mathsf{Mod_{LP}}(\alpha).$$

Let ω be an $\mathsf{LP_m}$ interpretation and K a flat knowledge base. We use $\omega!(K)$ to denote the set of variables of K assigned to B by ω, i.e.,

$$\omega!(K) = \{x \in \mathsf{Var}(K) \mid \omega(x) = \mathsf{B}\},$$

where $\mathsf{Var}(K)$ is the set of variables of K.

Based on $\omega!(K)$, the set of minimal models of K w.r.t. $\omega!(K)$ is given as follows:

$$\mathsf{MinMod_{LP}}(K) = \{\omega \in \mathsf{Mod_{LP}}(K) \mid \text{for all } \omega' \in \mathsf{Mod_{LP}}(K),\ |\omega'!(K)| \geqslant |\omega!(K)|\}.$$

Example 2.1. Consider $K = \{a, \neg a \vee b, \neg b \wedge c\}$ and $K' = \{a, \neg a, b, \neg b, c\}$. Then

$$\mathsf{MinMod_{LP}}(K) = \{\omega_1, \omega_2\}, \qquad \mathsf{MinMod_{LP}}(K') = \{\omega_3\},$$

where ω_1, ω_2, and ω_3 are given as follows:

	a	b	c
ω_1	T	B	T
ω_2	B	F	T
ω_3	B	B	T

The normalized minimum number of variables assigned inconsistent truth values in $\mathsf{LP_m}$ models of a knowledge base has been considered as an inconsistency measure for the knowledge base in [13,1]. Moreover, as stated in [13,1], such a measure is a special case of the assessments for the degree of contradiction presented in [22], which is based on the minimum cost of test actions that ensures a consistency recovery in K.

Definition 2.1 *($\mathsf{LP_m}$ Inconsistency Measure).* (See [1,13].) Let K be a knowledge base. The $\mathsf{I_{LP_m}}$ inconsistency measure for K, denoted $\mathsf{I_{LP_m}}(K)$, is defined as

$$\mathsf{I_{LP_m}}(K) = \frac{\min_{\omega \in \mathsf{Mod_{LP}}(K)}\{|\omega!(K)|\}}{|\mathcal{P}|}.$$

Note that $\mathcal{P}$ is always replaced with $\mathsf{Var}(K)$ for any individual knowledge base given in examples [13].

Example 2.2. Consider $K = \{a, \neg a \vee b, \neg b \wedge c\}$ and $K' = \{a, \neg a, b, \neg b, c\}$ again. Then

$$\mathsf{I_{LP_m}}(K) = \frac{1}{3}, \qquad \mathsf{I_{LP_m}}(K') = \frac{2}{3}.$$

Furthermore, we use $\mathsf{C_{LP_m}}(K)$ to denote the non-normalized version of $\mathsf{I_{LP_m}}(K)$, i.e.,

$$\mathsf{C_{LP_m}}(K) = \min_{\omega \in \mathsf{Mod_{LP}}(K)} \{|\omega!(K)|\}.$$

It is exactly the degree of contradiction of K presented in the case of the Priest's Logic of paradox [22].

On the other hand, minimal inconsistent subsets are also used to capture the inconsistency in a flat knowledge base [16,13,17,18]. A subset K' of K, is called a minimal inconsistent subset of K if K' is inconsistent, and none of its proper subsets

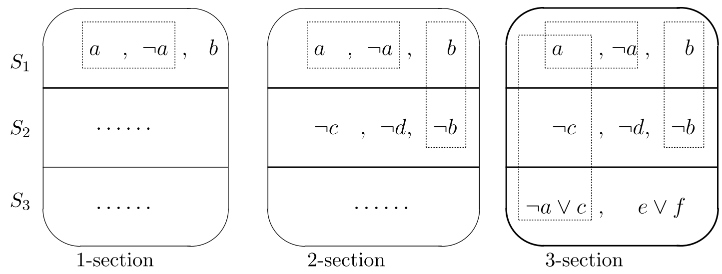

Fig. 1. Inconsistencies in each section.

is inconsistent. A formula of K is called a *free formula* if it does not belong to any minimal inconsistent subsets of K. We use $\mathsf{MI}(K)$ to denote the set of minimal inconsistent subsets of K, $\mathsf{MI}(K) = \{K' \subset K | K' \vdash \bot, \forall K'' \subset K', K'' \nvdash \bot\}$. The *MI* inconsistency measure presented in [16] considers the number of minimal inconsistent subsets of a knowledge base as an assessment for the inconsistency in the base.

Definition 2.2 *(MI inconsistency measure [16]).* Let K be a flat knowledge base. The MI inconsistency measure for K, denoted $\mathsf{I}_{\mathsf{MI}}(K)$, is defined as

$$\mathsf{I}_{\mathsf{MI}}(K) = \left|\mathsf{MI}(K)\right|.$$

Example 2.3. Consider $K = \{a, \neg a \vee b, \neg b \wedge c\}$ and $K' = \{a, \neg a, b, \neg b, c\}$ again. Then

$$\mathsf{MI}(K) = \{K\}, \qquad \mathsf{MI}\left(K'\right) = \left\{\{a, \neg a\}, \{b, \neg b\}\right\}.$$

So,

$$\mathsf{I}_{\mathsf{MI}}(K) = 1, \qquad \mathsf{I}_{\mathsf{MI}}\left(K'\right) = 2.$$

3. Multi-section inconsistency measures

How to articulate inconsistency in a stratified knowledge base has not yet been given much attention in the literature. Intuitively, the preference levels of formulas involved in inconsistency should be taken into account in establishing approaches to measuring inconsistency for stratified knowledge bases. We will propose two approaches to measuring inconsistency for stratified knowledge bases in this and the next section, respectively. The first approach aims to articulate inconsistencies occurring in each section of a stratified knowledge base. The second focuses on assessing inconsistency in a stratified knowledge base from a global perspective. Both the two approaches adopt vectorial measures to capture the inconsistency in a stratified knowledge base. In particular, the place of each component in a vectorial measure corresponds to one preference level, and then such vectorial measures are appropriate for characterizing inconsistency in a stratified knowledge base.

On the other hand, in some applications, a single value-based measure is more attractive than vectorial measures. For the second approach, we also provide a weighted version to capture the inconsistency in a stratified knowledge base by a single value. However, as we illustrated later, the weighted inconsistency measure fails to possess some expected properties.

As mentioned earlier, given a stratified knowledge base with n strata, formulas in the first i strata are more preferred than the last $n - i$ strata for each $1 \leqslant i < n$. This provides a usual way to separate a stratified knowledge base into two parts at each preference level i, i.e., the i-section and the formulas strictly less preferred than it. Moreover, these sections are also meaningful for resolving inconsistency in some applications. For example, consider a stratified knowledge base $(S_1 = \{a, \neg a, b\}, S_2 = \{\neg c, \neg d, \neg b\}, S_3 = \{\neg a \vee c, e \vee f\})$ illustrated in Fig. 1. Intuitively, the conflict between a and $\neg a$ is the unique focus in inconsistency measuring, moreover, b has nothing to do with inconsistency in case that we only focus on the most preferred formulas. In contrast, if we focus the first two strata, then b is also involved in inconsistency. Furthermore, we may find that a is involved again in the new inconsistency we meet at the third stratum.

Allowing for this, we need to identify where (at which stratum) inconsistencies occur and how severe these inconsistencies are in such cases. That is, an inconsistency measure for a stratified knowledge base should articulate the inconsistency in each section as well as in the whole knowledge base. On the other hand, for each i-section, the k-section of the whole base is also its k-section for all $1 \leqslant k \leqslant i$. Then it is natural to define such a framework for measuring inconsistency for stratified knowledge bases in the following way.

Definition 3.1 *(Multi-section inconsistency measure).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base and $\mathrm{Inc}_{\mathrm{m}}$ an inconsistency measure for stratified knowledge bases. Then $\mathrm{Inc}_{\mathrm{m}}(K)$ is called a multi-section inconsistency measure (MSIM for short) for K if

$$\mathrm{Inc_m}(K) = (c_1, c_2, \ldots, c_n)$$

such that

$$\mathrm{Inc_m}(K_{1\to i}) = (c_1, c_2, \ldots, c_i) \quad \text{for all } 1 \leqslant i \leqslant n.$$

Informally speaking, $\mathrm{Inc_m}(K)$ uses a vector instead of a single value to measure the conflicts in K such that the first i components of the vector together describe the conflicts in the i-section of K. This implies that the multi-section inconsistency measure allows us to characterize conflicts in a stratified knowledge at each preference level or stratum.

Next we introduce two kinds of instances of the multi-section inconsistency measure, i.e., the naive MSIM and the stratum-centric MSIM.

3.1. The naive MSIM

Given a stratified knowledge base, the i-cut of the base consists of formulas in the first i strata of the base (i.e., all formulas with at least the i-th preference level). We may consider using all the measures for the degree of inconsistency in each cut of the i-section together to assess inconsistencies in the section. We call such inconsistency measures naive MSIMs because they do not consider the total preorder relation on formulas within each i-cut.

Definition 3.2 *(Naive MSIM).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base and I an inconsistency measure for flat knowledge bases. A naive MSIM for K, denoted $\mathrm{Inc_m^I}(K)$, is defined as

$$\mathrm{Inc_m^I}(K) = (c_1, c_2, \ldots, c_n),$$

where $c_i = I(K(i))$ for all $1 \leqslant i \leqslant n$.

The naive MSIM for a stratified knowledge base essentially focuses on assessing the inconsistency in each i-cut of the knowledge base. That is, the naive MSIM captures the degree of contradiction among formulas we meet at each stratum. Note that

$$\mathrm{Inc_m^I}(K_{1\to i}) = (c_1, c_2, \ldots, c_i)$$

for all $1 \leqslant i \leqslant n$. So, it is a kind of multi-section inconsistency measure.

Now we give the following naive MSIM based on $\mathsf{C}_{\mathsf{LP_m}}$.

Definition 3.3 *($\mathsf{LP_m}$ MSIM).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base. Then the $\mathsf{LP_m}$ MSIM for K, denoted $\mathrm{Inc_m^{C_{LP}}}(K)$, is defined as

$$\mathrm{Inc_m^{C_{LP}}}(K) = (c_1, c_2, \ldots, c_n),$$

where $c_i = \mathsf{C}_{\mathsf{LP_m}}(K(i))$ for all $1 \leqslant i \leqslant n$.

Example 3.1. Consider the following stratified knowledge bases:

$$K_1 = \big(\{a\}, \{b\}, \{\neg a, \neg b\}\big), \qquad K_2 = \big(\{a\}, \{b\}, \{\neg a, c\}\big), \qquad K_3 = \big(\{a\}, \{b\}, \{c, \neg b\}\big),$$
$$K_4 = \big(\{a, \neg a\}, \{b\}, \{\neg b\}\big), \qquad K_5 = \big(\{a\}, \{\neg a, b\}, \{\neg b\}\big).$$

Then

$$\mathrm{Inc_m^{C_{LP}}}(K_1) = (0, 0, 2), \qquad \mathrm{Inc_m^{C_{LP}}}(K_2) = \mathrm{Inc_m^{C_{LP}}}(K_3) = (0, 0, 1),$$
$$\mathrm{Inc_m^{C_{LP}}}(K_4) = (1, 1, 2), \qquad \mathrm{Inc_m^{C_{LP}}}(K_5) = (0, 1, 2).$$

Moreover,

$$\mathrm{Inc_m^{C_{LP}}}(K_2) = \mathrm{Inc_m^{C_{LP}}}(K_3) < \mathrm{Inc_m^{C_{LP}}}(K_1) < \mathrm{Inc_m^{C_{LP}}}(K_5) < \mathrm{Inc_m^{C_{LP}}}(K_4).$$

From $\mathrm{Inc_m^{C_{LP}}}(K_1) = (0, 0, 2)$, we can get

$$\mathrm{Inc_m^{C_{LP}}}(K_{1_{1\to 1}}) = (0), \quad \text{and} \quad \mathrm{Inc_m^{C_{LP}}}(K_{1_{1\to 2}}) = (0, 0).$$

This means that both the 1-section $(\{a\})$ and the 2-section $(\{a\}, \{b\})$ of K_1 are consistent, and we meet inconsistency at last stratum. In contrast, $\mathrm{Inc_m^{C_{LP}}}(K_4) = (1, 1, 2)$ means that the first stratum of K_4 is inconsistent. In this sense, $\mathrm{Inc_m^{C_{LP}}}(K_1) < \mathrm{Inc_m^{C_{LP}}}(K_4)$ is intuitive.

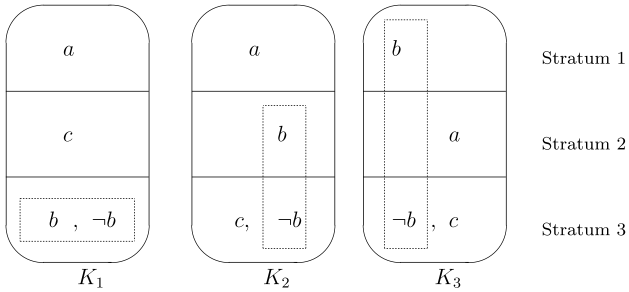

Fig. 2. Illustration of inconsistencies in different stratified knowledge bases.

On the other hand, $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K_1) = (0, 0, 2)$ means that both the atoms a and b are involved in the inconsistency we meet at last stratum of K_1, whilst $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K_2) = (0, 0, 1)$ means that only one atom is involved in the inconsistency we meet at the last stratum of K_2. So, it is also intuitive that $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K_2) < \mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K_1)$.

Similarly, we may use other inconsistency measures for flat knowledge bases such as I_{MI} to instantiate the naive MSIM.

3.2. Stratum-centric MSIM

The naive MSIM focuses on describing conflicts in a stratified knowledge base by characterizing how each cut of the base inconsistent is. On the other hand, the naive MSIM does not consider the preorder relation between formulas within each i-cut. Then the impact of preference on the inconsistency assessment has been not yet reflected intuitively within each i-cut. To illustrate this, consider three inconsistent stratified knowledge bases with the commensurable scale $K_1 = (\{a\}, \{c\}, \{b, \neg b\})$, $K_2 = (\{a\}, \{b\}, \{\neg b, c\})$, and $K_3 = (\{b\}, \{a\}, \{\neg b, c\})$. Note that the 2-cut of each stratified knowledge base is consistent, moreover, $K_1(3) = K_2(3) = K_3(3)$. Then $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}(K_1) = \mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}(K_2) = \mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}(K_3)$ always holds. However, as illustrated by Fig. 2, conflicts in the three knowledge bases are very different from each other. Note that different levels of stratum are involved in inconsistencies in the three knowledge bases. In detail, the formula $\neg b$ with the lowest preference level contradicts the most preferred formula b in K_3 and b in the second stratum of K_2, respectively, whilst only the formulas with the lowest preference level are involved in the inconsistency of K_1. Intuitively, K_3 is the most inconsistent one among the three stratified knowledge bases.

To address this, next we introduce a new multi-section inconsistency measure, which is based on strata involved in inconsistency in a stratified knowledge base. We start with minimal inconsistent subsets due to each stratum.

Definition 3.4 *(Minimal inconsistent subsets due to the i-th stratum).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base. The Minimal inconsistent subsets due to S_i, denoted $\mathrm{Mis}(S_i)$, is defined as

$$\mathrm{Mis}(S_1) = \mathsf{MI}(S_1),$$
$$\mathrm{Mis}(S_i) = \mathsf{MI}\big(K(i)\big) \setminus \mathsf{MI}\big(K(i-1)\big) \quad \text{for all } 1 < i \leqslant n.$$

Example 3.2. Consider $K = (\{a, \neg a\}, \{b\}, \{\neg b\})$. Then $\mathsf{MI}(K(1)) = \mathsf{MI}(K(2)) = \{M_1\}$ and $\mathsf{MI}(K(3)) = \{M_1, M_2\}$, where

$$M_1 = \{a, \neg a\}, \qquad M_2 = \{b, \neg b\}.$$

So,

$$\mathrm{Mis}(S_1) = \{M_1\}, \qquad \mathrm{Mis}(S_2) = \emptyset, \qquad \mathrm{Mis}(S_3) = \{M_2\}.$$

Note that $\mathrm{Mis}(S_2) = \emptyset$ means that S_2 brings no new minimal inconsistent subset of the 2-cut of K. It accords with our intuition.

In the above example, $M_2 = \{b, \neg b\}$ is the only minimal inconsistent subset relevant to S_3. Moreover, S_2 is the most preferred stratum involved in M_2. Then we can use the level number 2 to characterize the degree of inconsistency partially caused by S_3. In general, we can formalize this notion as follows.

Given a stratified KB $K = (S_1, S_2, \ldots, S_n)$ and a fixed level i with $\mathrm{Mis}(S_i) \neq \emptyset$ $(1 \leqslant i \leqslant n)$, we say that the k-th stratum S_k $(k \leqslant i)$ is involved in inconsistency due to S_i if $S_k \cap M \neq \emptyset$ for some $M \in \mathrm{Mis}(S_i)$. We use $\kappa(S_i)$ to denote the minimum level of strata involved in minimal inconsistent subsets due to S_i. For convenience, we set $\kappa(S_i) = i + 1$ if $\mathrm{Mis}(S_i) = \emptyset$. Formally,

$$\kappa(S_i) = \begin{cases} \min\{1 \leqslant k \leqslant i \mid \exists M \in \mathrm{Mis}(S_i) \text{ s.t. } S_k \cap M \neq \emptyset\}, & \text{if } \mathrm{Mis}(S_i) \neq \emptyset; \\ i + 1, & \text{otherwise.} \end{cases}$$

Recall the three knowledge bases illustrated in Fig. 2, then $\kappa(S_3) = 3$ for K_1, $\kappa(S_3) = 2$ for K_2, and $\kappa(S_3) = 1$ for K_3. It can make a distinction among the three knowledge bases.

We can use the number $\kappa(S_i)$ to characterize the inconsistency due to S_i. However, for technical reasons, we introduce the following multi-section inconsistency measure.

Definition 3.5 *(Stratum-centric MSIM).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base. The stratum-centric MSIM for K, denoted $\mathrm{Inc}^s_m(K)$, is defined as

$$\mathrm{Inc}^s_m(K) = (c_1, c_2, \ldots, c_n),$$

where $c_i = (i + 1 - \kappa(S_i))$ for all $1 \leqslant i \leqslant n$.

Note that $\kappa(S_i)$ captures the level of the most preferred stratum involved in inconsistency due to S_i, and $i + 1$ is a designated value for the case that S_i brings no new minimal inconsistent subsets. Then the more preferred strata involved in inconsistency due to S_i, the greater c_i is. In this sense, it provides an assessment for the level or severity of inconsistency due to S_i. Moreover,

$$\mathrm{Inc}^s_m(K_{1\to i}) = \mathrm{Inc}^s_m(K_{1\to i-1}) \uplus \big(i + 1 - \kappa(S_i)\big)$$

for all $i > 1$. Then the stratum-centric MSIM for a stratified knowledge base describes the most preferred stratum involved in inconsistency we meet at each stratum in such a way.

Example 3.3. Consider stratified knowledge bases in Example 3.1 again. Then

$$\mathrm{Inc}^s_m(K_1) = (0, 0, 3), \qquad \mathrm{Inc}^s_m(K_2) = (0, 0, 3), \qquad \mathrm{Inc}^s_m(K_3) = (0, 0, 2),$$
$$\mathrm{Inc}^s_m(K_4) = (1, 0, 2), \qquad \mathrm{Inc}^s_m(K_5) = (0, 2, 2).$$

Moreover,

$$\mathrm{Inc}^s_m(K_3) < \mathrm{Inc}^s_m(K_2) = \mathrm{Inc}^s_m(K_1) < \mathrm{Inc}^s_m(K_5) < \mathrm{Inc}^s_m(K_4).$$

Recall that the naive MSIM above cannot make a distinct between K_2 and K_3. However, Inc^s_m captures the intuition that the most preferred stratum involved in conflicts we meet at the last stratum in K_2 is more preferred than that in K_3.

Next we introduce some evident observations about Inc^s_m.

Proposition 3.1. *Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base and $\mathrm{Inc}^s_m(K) = (c_1, c_2, \ldots, c_n)$. Then*

(1) *$c_1 = 0$ if and only if S_1 is consistent, i.e., $\mathsf{MI}(S_1) = \emptyset$.*
(2) *$c_i = 0$ if and only if $\mathsf{MI}(K(i)) = \mathsf{MI}(K(i-1))$ for all $i > 1$.*
(3) *$c_i = 1$ if and only if $\forall M \in \mathrm{Mis}(S_i)$, $M \subseteq S_i$.*
(4) *$c_i > 1$ if and only if $\exists M \in \mathrm{Mis}(S_i)$ s.t. $M \cap S_{i+1-c_i} \neq \emptyset$ for all $i > 1$.*

Proof. We just provide a proof for (2). The other proofs are evident and similar to this one.

(2) Note that for $i > 1$, $c_i = 0. \Leftrightarrow \kappa(S_i) = i + 1. \Leftrightarrow \mathrm{Mis}(S_i) = \emptyset. \Leftrightarrow \mathsf{MI}(K(i)) = \mathsf{MI}(K(i-1))$. $\square$

Clearly, $c_i = 0$ means that minimal inconsistent subsets of $K(i)$ have nothing to do with S_i, i.e., S_i brings no new conflicts in $K_{1\to i}$ for each i. In contrast, $c_i = 1$ means that new conflicts $K_{1\to i}$ due to S_i are exactly conflicts between formulas in S_i, whilst $c_i > 1$ states that there are formulas in the more preferred stratum S_{i+1-c_i} are involved in new conflicts due to S_i. Moreover, the larger c_i is, the more preferred stratum involved in inconsistency due to S_i. In this sense, these observations show that each c_i of $\mathrm{Inc}^s_m(K)$ captures the new conflicts in $K_{1\to i}$ due to S_i within the context of K indeed.

We use the following example to illustrate the role of multi-section in characterizing inconsistency in a stratified knowledge base.

Example 3.4. Consider $K = (\{a\}, \{\neg a \vee b\}, \{\neg b, c\}, \{d, e\}, \{\neg d\})$. Then

$$\mathrm{Inc}^s_m(K) = (0, 0, 3, 0, 2).$$

By this multi-section measure, we can get the inconsistency measure for each section of K. For example, we can get

$$\mathrm{Inc}^s_m(K_{1\to 2}) = (0, 0), \qquad \mathrm{Inc}^s_m(K_{1\to 3}) = (0, 0, 3), \qquad \mathrm{Inc}^s_m(K_{1\to 4}) = (0, 0, 3, 0).$$

From these measures, we can conclude that the first two strata of K is consistent, but the third stratum brings conflicts to the base, moreover, the first stratum is involved in the conflicts. This implies that if we need to focus on preferred and consistent parts of the base, then the 2-section is a good choice.

On the other hand, the fourth stratum brings no new conflicts to the 4-section of the base, i.e., the conflicts in the 4-section arise from the first three strata, i.e., is the same as that arising from the 3-section.

4. Preference-based inconsistency measures

The multi-section inconsistency measure for a stratified knowledge base focuses on characterizing inconsistency in each section of that base. A complete assessment for the inconsistency in a stratified knowledge base consists of all the characterizations of inconsistency in sections. In some applications we also need to assess the inconsistency in a stratified knowledge base as a whole. In other words, we hope to use an inconsistency measure to capture all the conflicts in each stratum in an integrated way. Intuitively, such an inconsistency measure should take the preference information between different strata into account as well as the inconsistency in each stratum.

Given a flat knowledge base, the $\mathsf{LP_m}$ inconsistency measure focuses on counting the minimal number of inconsistent variables rather than inconsistent formulas under $\mathsf{LP_m}$ models of the base [1]. However, in a stratified knowledge base, its stratum-based structure induces a preference relation on formulas. So the inconsistency measure would be more precise and more natural if the preference information can combined when we count the number of formulas in each stratum that are interpreted inconsistent under a $\mathsf{LP_m}$ model. In order to define such an inconsistency measure, we first introduce the *rank* of a $\mathsf{LP_m}$ model.

Definition 4.1 *(Rank of a model).* Let $K = (S_1, S_2, \dots, S_n)$ be a stratified knowledge base and $\mathsf{Mod_{LP}}(K)$ the set of $\mathsf{LP_m}$ models of K. Given $\omega \in \mathsf{Mod_{LP}}(K)$, the rank of ω w.r.t. K, denoted $\mathrm{r_K}(\omega)$, is defined as

$$\mathrm{r_K}(\omega) = (r_1, r_2, \dots, r_n),$$

where $r_i = |\{\alpha \mid \alpha \in S_i, \omega(\alpha) = \mathsf{B}\}|$ for $i = 1, \dots, n$.

When the knowledge base is clear from the context, $\mathrm{r_K}(\omega)$ is abbreviated as $\mathrm{r}(\omega)$.

Example 4.1. Consider $K = (\{a\}, \{\neg a \vee b\}, \{\neg b, c\})$. Then

$$\mathsf{Mod_{LP}}(K) = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6\},$$

where ω_i for $1 \leqslant i \leqslant 6$ are given as follows:

	a	b	c	$\neg a \vee b$	$\neg b$
ω_1	T	B	T	B	B
ω_2	T	B	B	B	B
ω_3	B	F	T	B	T
ω_4	B	F	B	B	T
ω_5	B	B	T	B	B
ω_6	B	B	B	B	B

So,

$$\mathrm{r}(\omega_1) = (0,1,1) < \mathrm{r}(\omega_2) = (0,1,2) < \mathrm{r}(\omega_3) = (1,1,0) < \mathrm{r}(\omega_4) = \mathrm{r}(\omega_5) = (1,1,1) < \mathrm{r}(\omega_6) = (1,1,2).$$

Given the above definition, we could use the following notion to measure the inconsistency of a stratified KB.

Definition 4.2 *($\mathsf{PLP_m}$ measure).* Let $K = (S_1, S_2, \dots, S_n)$ be a stratified knowledge base. The $\mathsf{PLP_m}$ measure for K, denoted $\mathrm{Inc_{LP_m}}(K)$, is defined as

$$\mathrm{Inc_{LP_m}}(K) = \min\bigl\{\mathrm{r}(\omega) \bigm| \omega \in \mathsf{Mod_{LP}}(K)\bigr\}.$$

Intuitively, the $\mathsf{PLP_m}$ measure for a stratified knowledge base is the minimum rank of its $\mathsf{LP_m}$ models.

Example 4.2. Consider stratified knowledge bases in Example 3.1 again. Then

$$\mathrm{Inc_{LP_m}}(K_1) = (1,1,2), \quad \mathrm{Inc_{LP_m}}(K_2) = (1,0,1), \quad \mathrm{Inc_{LP_m}}(K_3) = (0,1,1),$$
$$\mathrm{Inc_{LP_m}}(K_4) = (2,1,1), \quad \mathrm{Inc_{LP_m}}(K_5) = (1,2,1).$$

Moreover,

$$\mathrm{Inc}_{\mathrm{LP_m}}(K_3) < \mathrm{Inc}_{\mathrm{LP_m}}(K_2) < \mathrm{Inc}_{\mathrm{LP_m}}(K_1) < \mathrm{Inc}_{\mathrm{LP_m}}(K_5) < \mathrm{Inc}_{\mathrm{LP_m}}(K_4).$$

The following example shows that the preference-based measure $\mathrm{Inc}_{\mathrm{LP_m}}$ is essentially different from multi-section inconsistency measures.

Example 4.3. Consider $K = (\{a\}, \{\neg a \vee b, c\}, \{\neg b, d\})$. Then

$$\mathrm{Inc}_{\mathrm{LP_m}}(K_{1\to 2}) = (0, 0), \quad \text{but } \mathrm{Inc}_{\mathrm{LP_m}}(K_{1\to 3}) = (0, 1, 1).$$

So,

$$\mathrm{Inc}_{\mathrm{LP_m}}(K_{1\to 3}) \neq \mathrm{Inc}_{\mathrm{LP_m}}(K_{1\to 2}) \uplus 1.$$

We must point out $\mathrm{PLP_m}$ measure $\mathrm{Inc}_{\mathrm{LP_m}}$ is not the lexicographical cardinality of minimal inconsistent subsets w.r.t. $\leqslant_K$. To illustrate this, consider the following example.

Example 4.4. Consider $K = (\{a\}, \{b\}, \{c\}, \{\neg c \wedge d \wedge \neg d\})$. Then

$$K' = \big(\{\}, \{\}, \{\}, \{\neg c \wedge d \wedge \neg d\}\big)$$

is the minimal inconsistent subset of K w.r.t. $\leqslant_K$. But for any $\mathrm{LP_m}$ model ω of K,

$$\omega(c) = \omega(\neg c \wedge d \wedge \neg d) = \mathsf{B}.$$

So,

$$\mathrm{Inc}_{\mathrm{LP_m}}(K) = (0, 0, 1, 1) > (0, 0, 0, 1) = \mathrm{lc}\big(K'\big).$$

Given a $\mathrm{LP_m}$ model of K, its rank describes the number of formulas assigned to inconsistent truth values in each stratum under that model. In some applications we are also interested in what proportion of formulas in each stratum involved in inconsistency. To address this issue, we use a normalized version of the $\mathrm{PLP_m}$ measure for a stratified knowledge base. To this end, we first introduce the normalized rank of a $\mathrm{LP_m}$ model.

Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base and $\mathrm{r}(\omega) = (r_1, r_2, \ldots, r_n)$ the rank of the model ω of K. Then the normalized rank of ω, denoted $\mathrm{r}^{\mathrm{N}}(\omega)$, is defined as

$$\mathrm{r}^{\mathrm{N}}(\omega) = \big(r_1^N, r_2^N, \ldots, r_n^N\big),$$

where $r_i^N = r_i/|S_i|$ for all $1 \leqslant i \leqslant n$.

Now we are ready to define our normalized version of the preference-based inconsistency measure $\mathrm{Inc}_{\mathrm{LP_m}}$ for stratified KB based on the normalized rank of a $\mathrm{LP_m}$ model.

Definition 4.3 *($\mathrm{NLP_m}$ measure).* Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base. The $\mathrm{NLP_m}$ measure for K, denoted $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$, is defined as

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K) = \min\big\{\mathrm{r}^{\mathrm{N}}(\omega) \,|\, \omega \in \mathsf{Mod}_{\mathsf{LP}}(K)\big\}.$$

Example 4.5. Consider stratified knowledge bases in Example 3.1 again. Then

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_1) = (1, 1, 1), \qquad \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_2) = \left(1, 0, \frac{1}{2}\right), \qquad \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_3) = \left(0, 1, \frac{1}{2}\right),$$
$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_4) = (1, 1, 1), \qquad \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_5) = (1, 1, 1).$$

Moreover,

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_3) < \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_2) < \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_1) = \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_4) = \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_5).$$

Note that within each of K_1, K_4, and K_5, all the formulas in each stratum are involved in inconsistency (i.e., assigned inconsistent truth value in $\mathrm{LP_m}$ models with the minimal normalized rank). In this sense, $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_1) = \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_4) = \mathrm{Inc}_{\mathrm{N_{LP_m}}}(K_5)$ is intuitive.

Roughly speaking, $\mathrm{Inc}_{\mathrm{LP_m}}(K)$ focuses on counting the number of formulas involved in inconsistency for each stratum of K, whilst the normalized version $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ aims to capture the ratio of the number of formulas of each stratum involved in inconsistency to the cardinality of the stratum. That is, the two measures capture different aspects of inconsistency in a stratified knowledge base. Then it is not surprised that $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ behaviors differently from $\mathrm{Inc}_{\mathrm{LP_m}}(K)$ in supporting different logical properties, as illustrated later.

Note that either multi-section inconsistency measures or the preference-based measures for a stratified knowledge base with n strata use n-size vectors to capture the inconsistency in the base. However, a single inconsistency value is often more attractive to many applications than vectorial inconsistency assessment. A usual way is to use a set of weights to articulate the preference relation on a stratified knowledge bases. Actually, the choice of weights is rather sensitive to application domains because it is difficult to assure that all the stratified knowledge bases are commensurate. Moreover, as illustrated later, the weight-based approach may fail to support some expected properties for characterizing inconsistency in stratified knowledge bases.

We start with the weights compatible with a stratified knowledge base. A vector $\vec{\mathrm{w}} = (w_1, w_2, \dots, w_n)$ of weights is compatible with a stratified knowledge base K with n strata, if $w_1 > w_2 > \cdots > w_n > 0$. Essentially, we provide no condition about $\vec{\mathrm{w}}$ other than linear order relation on $\vec{\mathrm{w}}$. It makes the choice of weights flexible in applications. Given an $\vec{\mathrm{w}}$ compatible with K, we may define the following weighted version of the $\mathrm{PLP_m}$ measure for K.

Definition 4.4 *($\mathrm{WLP_m}$ measure).* Let $K = (S_1, S_2, \dots, S_n)$ be a stratified knowledge base and $\vec{\mathrm{w}} = (w_1, w_2, \dots, w_n)$ a vector of weights compatible with K. The $\mathrm{WLP_m}$ measure under $\vec{\mathrm{w}}$ for K, denoted $\mathrm{Inc_w}(K)$, is defined as

$$\mathrm{Inc_w}(K) = \mathrm{Inc}_{\mathrm{LP_m}}(K) \cdot \vec{\mathrm{w}}^{\tau},$$

where $\vec{\mathrm{w}}^{\tau}$ is the transpose of $\vec{\mathrm{w}}$.

Example 4.6. Consider stratified knowledge bases in Example 3.1 again. Let $\vec{\mathrm{w}} = (w_1, w_2, w_3)$, then

$$\mathrm{Inc_w}(K_1) = w_1 + w_2 + 2w_3,$$
$$\mathrm{Inc_w}(K_2) = w_1 + w_3,$$
$$\mathrm{Inc_w}(K_3) = w_2 + w_3,$$
$$\mathrm{Inc_w}(K_4) = 2w_1 + w_2 + w_3,$$
$$\mathrm{Inc_w}(K_5) = w_1 + 2w_2 + w_3.$$

Moreover,

$$\mathrm{Inc_w}(K_3) < \mathrm{Inc_w}(K_2) < \mathrm{Inc_w}(K_1) < \mathrm{Inc_w}(K_5) < \mathrm{Inc_w}(K_4).$$

In particular, for $\vec{\mathrm{w}}_{\mathrm{f}} = (1, \frac{1}{2}, \frac{1}{3})$,

$$\mathrm{Inc}_{\mathrm{w_f}}(K_1) = \frac{13}{6}, \qquad \mathrm{Inc}_{\mathrm{w_f}}(K_2) = \frac{4}{3}, \qquad \mathrm{Inc}_{\mathrm{w_f}}(K_3) = \frac{5}{6}, \qquad \mathrm{Inc}_{\mathrm{w_f}}(K_4) = \frac{17}{6}, \qquad \mathrm{Inc}_{\mathrm{w_f}}(K_5) = \frac{11}{6}.$$

As shown by the following proposition, $\mathrm{Inc_w}$ is less discriminative than $\mathrm{Inc}_{\mathrm{LP_m}}$.

Proposition 4.1. *If* $\mathrm{Inc}_{\mathrm{LP_m}}(K) = \mathrm{Inc}_{\mathrm{LP_m}}(K')$, *then* $\mathrm{Inc_w}(K) = \mathrm{Inc_w}(K')$. *But the converse does not hold.*

Proof. We only provide the following counterexample for the converse. Consider $K = (\{a\}, \{b, \neg b\}, \{c, d, e\})$ and $K' = (\{a\}, \{b, c\}, \{d, \neg d, \neg d \wedge e\})$. Suppose that $\vec{\mathrm{w}}_{\mathrm{f}} = (1, \frac{1}{2}, \frac{1}{3})$. Then

$$\mathrm{Inc}_{\mathrm{w_f}}(K) = \mathrm{Inc}_{\mathrm{w_f}}\big(K'\big) = 1.$$

But

$$\mathrm{Inc}_{\mathrm{LP_m}}(K) = (0, 2, 0) > (0, 0, 3) = \mathrm{Inc}_{\mathrm{LP_m}}\big(K'\big). \qquad \Box$$

The aim of the weighted $\mathrm{PLP_m}$ measure is to integrate assessments for strata listed in lexicographical order into a single value. Such an integration based on weights may weaken the difference between preference levels of strata. This is why $\mathrm{Inc_w}$ is less discriminative than $\mathrm{Inc}_{\mathrm{LP_m}}$.

5. Logical properties

To our best knowledge, there is not yet a desirable set of properties designed for inconsistency measures in the case of stratified knowledge bases. However, how to characterize inconsistency measures is still a challenge even in the case of classical knowledge bases. Compared to a growing number of inconsistency measures, there are relatively fewer proposals for developing suitable properties for them. One reason is that it is difficult to describe inconsistent knowledge bases in classical logics due to the principle of explosion, and another one is the diversity of paraconsistent logics [30].

Mu et al. have presented that a desirable set of properties for an inconsistency measure should at least contain three aspects, i.e., constraints on the nature of inconsistency, constraints on the change of inconsistency, and special characteristics of the measure [18]. Roughly speaking, the first aspect requires that an inconsistency measure can articulate the nature of inconsistent knowledge bases, i.e., it is capable of distinguishing inconsistent knowledge bases from consistent knowledge bases. The second aspect focuses on whether the degree (or amount) of inconsistency in a knowledge base increases when the base is enlarged or strengthened logically. The last aspect is always associated with the form of inconsistency characterization (e.g., minimal inconsistent subsets or atoms assigned to inconsistent truth values) as well as the underlying structure of knowledge bases (e.g., stratified or not).

The property of *Consistency* presented in [1,16,13] is considered as one of the intuitive constraints on the nature of inconsistency. It states that any nonnegative inconsistency measure should assign zero to only consistent knowledge bases. The satisfaction of *Consistency* for a given measure ensures that the measure is indeed an inconsistency measure, because *Consistency* makes the inconsistency measure capable of distinguishing inconsistent knowledge bases from consistent knowledge bases [31]. Obviously, it is also suitable for inconsistency measures in the case of stratified knowledge bases. Let Inc be an inconsistency measure for stratified knowledge bases, we may express *Consistency* as follows:

- *Consistency*. $\mathrm{Inc}(K) = 0$ if and only if K is consistent.

Secondly, the property of *Monotony* used in [1,16,13] describes that the amount of inconsistency in a knowledge base cannot decrease when the base is enlarged by adding new formulas. In particular, the properties of *Free Formula Independence* and *Safe Formula Independence* used in [1,16,13] describe two cases that the amount of inconsistency in a knowledge base remains unchanged when the base is enlarged by adding formulas having nothing to do with some special kinds of inconsistency characterization, respectively. Intuitively, the three properties are also suitable for the case of stratified knowledge bases. We may express them as follows:

- *Monotony*. $\mathrm{Inc}(K) \leqslant \mathrm{Inc}(K \cup_i \{\alpha\})$.
- *Free Formula Independence*. $\mathrm{Inc}(K \cup_i \{\alpha\}) = \mathrm{Inc}(K)$ if α is a free formula of $K \cup_i \{\alpha\}$.
- *Safe Formula Independence*. $\mathrm{Inc}(K \cup_i \{\alpha\}) = \mathrm{Inc}(K)$ if $\mathrm{Var}(\{\alpha\}) \cap \mathrm{Var}(K) = \emptyset$ and $\alpha \nvdash \bot$.

Note that *Free Formula Independence* states that free formulas cannot affect the amount of inconsistency in a knowledge base when the inconsistency is characterized in the form of minimal inconsistent subsets, whilst *Safe Formula Independence* (also termed *Weak Independence* in [31]) states the any consistent formulas consisting of new atoms brings no new conflict in a knowledge base. In this sense, the former is appropriate only for characterizing inconsistency measures based on minimal inconsistent subsets.

In addition, the property of *Dominance* adopted in [1,16,13] states that logically stronger formulas bring more conflicts in a knowledge base. It is also intuitive in the case of stratified knowledge bases. Then we may express it as follows:

- *Dominance*. $\mathrm{Inc}(K \cup_i \{\alpha\}) \geqslant \mathrm{Inc}(K \cup_i \{\beta\})$ if $\alpha \vdash \beta$ and $\alpha \nvdash \bot$.

Note that the last four properties can be considered as constraints on the change of inconsistency. Roughly speaking, these five slightly adapted properties focus on describing some general characteristics of inconsistency measures for stratified knowledge bases. In other words, these properties do not consider the stratum-based structure or the preference relation over a stratified knowledge base explicitly. To address this, we introduce one new property, which reflects the impact of the priority level of a formula on the inconsistency assessment. It states that a given formula at more preferred stratum makes a stratified knowledge base more inconsistent that the same formula at less preferred stratum.

- *Inferior Stratum*. $\mathrm{Inc}(K \cup_i \{\alpha\}) \leqslant \mathrm{Inc}(K \cup_j \{\alpha\})$ if $i \geqslant j$.

As mentioned in [18], *Monotony*, *Dominance*, *Free Formula Independence*, and *Safe Formula Independence* are designed for only characterizing inconsistency measures which focus on pure inconsistency (formulas or atoms involved in inconsistency) in a knowledge base. However, as mentioned earlier, in some applications we need also some measures such as the $\mathsf{NLP_m}$ measure tell us what proportion of formulas or atoms involved in inconsistency. Generally, we call such measures normalized inconsistency measures. It has been shown that normalized inconsistency measures behave differently from other measures, and then none of four properties is suitable for such measures [18].

On the other hand, *Inferior Stratum* is also a property only suitable for the inconsistency measures focusing on pure inconsistency. To illustrate this, consider adding a new consistent formula to a stratum, it cannot bring any new conflict, but it can enlarge the stratum. This may decrease the degree of inconsistency (the proportion of inconsistent formulas) we meet at the stratum in some cases.

Allowing for this, we adopt the properties *Consistency*, *Monotony*, *Dominance*, *Safe Formula Independence*, and *Inferior Stratum* to characterize all the inconsistency measures for the amount of inconsistency. In addition, we also consider *Free Formula Independence* as a property for the inconsistency measures based on minimal inconsistent subsets.

Let us first see what is the relation between the naive MSIM for stratified knowledge bases and the inconsistency measure used in it.

Proposition 5.1. *Let* $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$ *be a naive MSIM and* I *the inconsistency measure used in* $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$. *Then*

(1) $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$ *satisfies* Consistency (*resp.* Monotony, Dominance, Free Formula Independence, Safe Formula Independence) *if and only if* I *satisfies* Consistency (*resp.* Monotony, Dominance, Free Formula Independence, Safe Formula Independence).
(2) $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$ *satisfies* Inferior Stratum *if* I *satisfies* Monotony.

Proof. (1) "⇒". Let S be a flat knowledge base. Consider $K = (S)$, then

$$\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}(K) = \big(I(S)\big).$$

Therefore, if $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$ satisfies *Consistency*(resp. *Monotony*, *Dominance*, *Free Formula Independence*, *Safe Formula Independence*), then I satisfies *Consistency* (resp. *Monotony*, *Dominance*, *Free Formula Independence*, *Safe Formula Independence*).

"⇐". Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base.

- Suppose that I satisfies *Consistency*, then

$$\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}(K) = (0, 0, \ldots, 0) \Leftrightarrow I\big(K(n)\big) = 0 \Leftrightarrow K \nvdash \bot.$$

  So, $\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}$ satisfies *Consistency*.
- Suppose that I satisfies *Monotony*. Note that

$$\big(K \cup_i \{\alpha\}\big)(k) = K(k)$$

  for all $k < i$, and

$$\big(K \cup_i \{\alpha\}\big)(i) = K(i) \cup \{\alpha\}.$$

  Then

$$I\big(\big(K \cup_i \{\alpha\}\big)(k)\big) = I\big(K(k)\big)$$

  for all $k < i$, and

$$I\big(\big(K \cup_i \{\alpha\}\big)(i)\big) = I\big(K(i) \cup \{\alpha\}\big) \geqslant I\big(K(i)\big).$$

  So,

$$\mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}\big(K \cup_i \{\alpha\}\big) \geqslant \mathrm{Inc}^{\mathrm{I}}_{\mathrm{m}}(K).$$

- If I satisfies *Dominance*. Let $\alpha \vdash \beta$ and $\alpha \nvdash \bot$. Note that

$$\big(K \cup_i \{\alpha\}\big)(k) = \big(K \cup_i \{\beta\}\big)(k)$$

  for all $k < i$, and

$$\big(K \cup_i \{\alpha\}\big)(i) = K(i) \cup \{\alpha\},$$

$$\big(K \cup_i \{\beta\}\big)(i) = K(i) \cup \{\beta\}.$$

  So,

$$I\big(\big(K \cup_i \{\alpha\}\big)(k)\big) = I\big(\big(K \cup_i \{\beta\}\big)(k)\big)$$

  for all $k < i$. But

$$I\big(\big(K \cup_i \{\alpha\}\big)(i)\big) = I\big(K(i) \cup \{\alpha\}\big) \geqslant I\big(K(i) \cup \{\beta\}\big) = I\big(\big(K \cup_i \{\beta\}\big)(i)\big)$$

because I satisfies *Dominance*. Therefore,

$$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}\big(K \cup_i \{\alpha\}\big) \geqslant \mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}\big(K \cup_i \{\beta\}\big).$$

- Suppose that I satisfies *Free Formula Independence*. Note that

$$\big(K \cup_i \{\alpha\}\big)(k) = K(k)$$

for all $k < i$, and

$$\big(K \cup_i \{\alpha\}\big)(j) = K(j) \cup \{\alpha\}$$

for all $j \geqslant i$. Let α be a free formula of $K \cup_i \{\alpha\}$, then

$$I\big(\big(K \cup_i \{\alpha\}\big)(m)\big) = I\big(K(m)\big)$$

for all $1 \leqslant m \leqslant n$. So,

$$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}\big(K \cup_i \{\alpha\}\big) = \mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}(K).$$

- The proof about *Safe Formula Independence* is very similar to that about *Free Formula Independence*.

(2) Let $K = (S_1, S_2, \ldots, S_n)$ and I an inconsistency measure satisfying *Monotony*. Suppose that $1 \leqslant i \leqslant j \leqslant n$. Note that

$$\big(K \cup_i \{\alpha\}\big)(k) = \big(K \cup_j \{\alpha\}\big)(k)$$

for all $k < i$, and

$$\big(K \cup_i \{\alpha\}\big)(i) = K(i) \cup \{\alpha\},$$

$$\big(K \cup_j \{\alpha\}\big)(i) = \begin{cases} K(i), & j > i \\ (K \cup_i \{\alpha\})(i), & j = i. \end{cases}$$

So,

$$I\big(\big(K \cup_i \{\alpha\}\big)(k)\big) = I\big(\big(K \cup_j \{\alpha\}\big)(k)\big)$$

for all $k < i$. But

$$I\big(\big(K \cup_i \{\alpha\}\big)(i)\big) = I\big(K(i) \cup \{\alpha\}\big) \geqslant I\big(\big(K \cup_j \{\alpha\}\big)(i)\big)$$

because I satisfies *Monotony*. Therefore,

$$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}\big(K \cup_i \{\alpha\}\big) \geqslant \mathrm{Inc}_{\mathrm{m}}^{\mathrm{I}}\big(K \cup_j \{\alpha\}\big) \quad \text{if } i \leqslant j. \qquad \square$$

It has been shown that $\mathsf{C}_{\mathsf{LP_m}}$ satisfies *Consistency*, *Monotony*,*Dominance* and *Safe Formula Independence* [13]. Evidently, we can get the following corollary from Proposition 5.1.

Corollary 5.1. *The naive MSIM* $\mathrm{Inc}_{\mathrm{m}}^{\mathsf{C_{LP}}}$ *satisfies* Consistency, Monotony, Dominance, Safe Formula Independence, *and* Inferior Stratum.

Concerning the second kind of the multi-section inconsistency measure $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$, we have identified the following properties.

Proposition 5.2. *The stratum-centric MSIM* $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$ *satisfies* Consistency, Monotony, Free Formula Independence, Safe Formula Independence, *and* Inferior Stratum.

Proof. Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base.

- *Consistency*. "$\Rightarrow$." $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K) = \vec{0}$ implies that for each S_i, $\kappa(S_i) = i + 1$. This means that each S_i brings no new conflict, i.e., each $K(i)$ is consistent. Particularly, $K(n)$ is consistent, that is, K is consistent.
"$\Leftarrow$." K is consistent, then each $K(i)$ is consistent. So, $\kappa(S_i) = i + 1$ for each i. Therefore, $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K) = \vec{0}$.

- *Monotony*. Let $K \cup_i \{\alpha\} = (T_1, T_2, \ldots, T_n)$. For $i > 1$, note that

$$\big(K \cup_i \{\alpha\}\big)(i-1) = K(i-1),$$

and

$$\big(K \cup_i \{\alpha\}\big)(j) = K(j) \cup \{\alpha\}$$

for all $j \geqslant i$. Then

$$\mathsf{MI}\big(\big(K \cup_i \{\alpha\}\big)(i-1)\big) = \mathsf{MI}\big(K(i-1)\big),$$

and

$$\mathsf{MI}\big(\big(K \cup_i \{\alpha\}\big)(j)\big) \supseteq \mathsf{MI}\big(K(j)\big).$$

So,

$$\mathrm{Mis}(S_j) \subseteq \mathrm{Mis}(T_j).$$

Then, by the definition of κ,

$$\kappa(S_j) \geqslant \kappa(T_j).$$

Further,

$$i + 1 - \kappa(S_j) \leqslant i + 1 - \kappa(T_i).$$

So, $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K) \leqslant \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K \cup_i \{\alpha\})$.
Then for $K \cup_1 \{\alpha\}$, $T_1 = S_1 \cup \{\alpha\}$, and $T_j = S_j$ for each $2 \leqslant j \leqslant n$. Note that $\mathsf{MI}(S_1) \subseteq \mathsf{MI}(T_1)$. So, for $k \geqslant 1$,

$$\mathrm{Mis}(S_k) \subseteq \mathrm{Mis}(T_k).$$

Now we can get

$$\kappa(S_k) \geqslant \kappa(T_k).$$

Evidently, $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K) \leqslant \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K \cup_1 \{\alpha\})$ holds.
- *Free Formula Independence*. Let α be a free formula of $K \cup_i \{\alpha\}$ and $K \cup_i \{\alpha\} = (T_1, T_2, \ldots, T_n)$. Then for all $1 \leqslant k \leqslant n$, $\mathsf{MI}((K \cup_i \{\alpha\})(k)) = \mathsf{MI}(K(k))$. So, $\kappa(S_j) = \kappa(T_j)$. Therefore,

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K) = \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_i \{\alpha\}\big).$$

- *Safe Formula Independence*. If $\mathrm{Var}(\{\alpha\}) \cap \mathrm{Var}(K) = \emptyset$ and $\alpha \nvdash \bot$, then α is a free formula of $K \cup_i \{\alpha\}$. From *Free Formula Independence*,

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K) = \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_i \{\alpha\}\big).$$

- *Inferior Stratum*. Note that for $k < j$,

$$\big(K \cup_i \{\alpha\}\big)(k) = \big(K \cup_j \{\alpha\}\big)(k) = K(k);$$

for $j \leqslant k < i$,

$$\big(K \cup_j \{\alpha\}\big)(k) = \big(K \cup_i \{\alpha\}\big)(k) \cup \{\alpha\} = K(k) \cup \{\alpha\};$$

and for $i \leqslant k \leqslant n$,

$$\big(K \cup_i \{\alpha\}\big)(k) = \big(K \cup_j \{\alpha\}\big)(k) = K(k) \cup \{\alpha\}.$$

For the simplicity, let $K \cup_j \{\alpha\} = (T_1, T_2, \ldots, T_n)$ and $K \cup_i \{\alpha\} = (R_1, R_2, \ldots, R_n)$. So,

$$\kappa(S_k) = \kappa(T_k) = \kappa(R_k)$$

for $k < j$, and

$$\kappa(T_k) \leqslant \kappa(S_k) = \kappa(R_k)$$

for $j \leqslant k < i$, and

$$\kappa(T_k) \leqslant \kappa(R_k) \leqslant \kappa(S_k)$$

for $k \geqslant i$. Therefore,

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_i \{\alpha\}\big) \leqslant \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_j \{\alpha\}\big) \quad \text{if } i \geqslant j. \qquad \square$$

Also, we have obtained that $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}$ fails to support the property of *Dominance*.

Proposition 5.3. $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}$ *does not satisfy* Dominance.

Proof. Consider the following counterexample: Let $K = (\{\neg a \vee b\}, \{c\}, \{\neg b\})$. Then

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_2 \{a\}\big) = (0, 0, 3) > \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_2 \big\{a \wedge (\neg a \vee b)\big\}\big) = (0, 0, 2). \qquad \square$$

However, as analyzed in [18], such a failure is largely due to the syntax sensitivity of minimal inconsistent subsets. Now we use the following example to illustrate the change of inconsistency in a knowledge base.

Example 5.1. Let $K = (\{a\}, \{b\}, \{c\})$. Consider the following changes for K:

$$K \cup_1 \{d\} = \big(\{a, d\}, \{b\}, \{c\}\big),$$
$$K \cup_2 \{\neg b\} = \big(\{a\}, \{b, \neg b\}, \{c\}\big),$$

and

$$K \cup_3 \{\neg b\} = \big(\{a\}, \{b\}, \{c, \neg b\}\big).$$

Note that d is a free (safe) formula of $K \cup_1 \{d\}$. Evidently,

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_2 \{\neg b\}\big) = (0, 1, 0) > \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_3 \{\neg b\}\big) = (0, 0, 2) > \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\big(K \cup_1 \{d\}\big) = (0, 0, 0) = \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K).$$

The results accord with *Free* (*Safe*) *Formula Independence*, *Monotony*, and *Inferior Stratum*, as well as *Consistency*.

We have obtained that the $\mathsf{PLP_m}$ measure $\mathrm{Inc}_{\mathrm{LP_m}}$ possesses all the five properties.

Proposition 5.4. *The* $\mathsf{PLP_m}$ *measure* $\mathrm{Inc}_{\mathrm{LP_m}}$ *satisfies* Consistency, Monotony, Dominance, Safe Formula Independence, *and* Inferior Stratum.

Proof. Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base.

- *Consistency*. $\mathrm{Inc}_{\mathrm{LP_m}}(K) = \vec{0}$. $\Leftrightarrow$ There exists at least one model ω of K s.t. $\forall \alpha \in K$, $\omega(\alpha) \neq \mathsf{B}$. $\Leftrightarrow$ K is consistent.
- *Monotony*. Note that $\mathsf{Mod_{LP}}(K \cup_i \{\alpha\}) \subseteq \mathsf{Mod_{LP}}(K)$. $\forall \omega \in \mathsf{Mod_{LP}}(K \cup_i \{\alpha\})$, we use $\mathrm{r}_{K\cup_i\{\alpha\}}(\omega)$ and $\mathrm{r}_K(\omega)$ to denote the ranks of ω in $K \cup_i \{\alpha\}$ and K, respectively. Then

  $$\mathrm{r}_K(\omega) \leqslant \mathrm{r}_{K\cup_i\{\alpha\}}(\omega)$$

  because

  $$\big\{\beta \in S_i \,\big|\, \omega(\beta) = \mathsf{B}\big\} \subseteq \big\{\beta \in S_i \cup \{\alpha\} \,\big|\, \omega(\beta) = \mathsf{B}\big\}.$$

  Then

  $$\min\big\{\mathrm{r}_K(\omega) | \omega \in \mathsf{Mod_{LP}}(K)\big\} \leqslant \min\big\{\mathrm{r}_{K\cup_i\{\alpha\}}(\omega) \,\big|\, \omega \in \mathsf{Mod_{LP}}\big(K \cup_i \{\alpha\}\big)\big\}.$$

  That is,

  $$\mathrm{Inc}_{\mathrm{LP_m}}(K) \leqslant \mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big).$$
- *Dominance*. Note that $\mathsf{Mod_{LP}}(K \cup_i \{\alpha\}) \subseteq \mathsf{Mod_{LP}}(K \cup_i \{\beta\})$. $\forall \omega \in \mathsf{Mod_{LP}}(K \cup_i \{\alpha\})$, we use $\mathrm{r}_{K\cup_i\{\alpha\}}(\omega)$ and $\mathrm{r}_{K\cup_i\{\beta\}}(\omega)$ to denote the ranks of ω in $K \cup_i \{\alpha\}$ and $K \cup_i \{\beta\}$, respectively. Then

  $$\mathrm{r}_{K\cup_i\{\beta\}}(\omega) \leqslant \mathrm{r}_{K\cup_i\{\alpha\}}(\omega)$$

  because

  $$\big|\big\{\gamma \in S_i \cup \{\beta\} \,\big|\, \omega(\gamma) = \mathsf{B}\big\}\big| \leqslant \big|\big\{\gamma \in S_i \cup \{\alpha\} \,\big|\, |\omega(\gamma) = \mathsf{B}\big\}\big|.$$

  Then

  $$\min\big\{\mathrm{r}_{K\cup_i\{\beta\}}(\omega) \,\big|\, \omega \in \mathsf{Mod_{LP}}\big(K \cup_i \{\beta\}\big)\big\} \leqslant \min\big\{\mathrm{r}_{K\cup_i\{\alpha\}}(\omega) \,\big|\, \omega \in \mathsf{Mod_{LP}}\big(K \cup_i \{\alpha\}\big)\big\}.$$

  That is,

  $$\mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\beta\}\big) \leqslant \mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big).$$

- *Safe Formula Independence*. Suppose that $\mathrm{Var}(\{\alpha\}) \cap \mathrm{Var}(K) = \emptyset$ and $\alpha \nvdash \bot$. From the proof of *Monotony*, we get that

  $$\mathsf{Mod}_{\mathsf{LP}}\big(K \cup_i \{\alpha\}\big) \subseteq \mathsf{Mod}_{\mathsf{LP}}(K).$$

  Further, note that $\mathrm{Var}(\{\alpha\}) \cap \mathrm{Var}(K) = \emptyset$ and $\alpha \nvdash \bot$, then

  $$\omega(\alpha) \neq \mathsf{B}$$

  for all $\omega \in \mathsf{Mod}_{\mathsf{LP}}(K)$ s.t. $\mathrm{r}_K(\omega) = \min\{\mathrm{r}_K(\omega') | \omega' \in \mathsf{Mod}_{\mathsf{LP}}(K)\}$. If $\omega(\alpha) = \mathsf{T}$, then

  $$\omega \in \mathsf{Mod}_{\mathsf{LP}}\big(K \cup_i \{\alpha\}\big).$$

  If $\omega(\alpha) = \mathsf{F}$, then we can get ω' by changing only atoms in α such that

  $$\forall \beta \in K, \quad \omega'(\beta) = \omega(\beta) \text{ and } \omega'(\alpha) = \mathsf{T}.$$

  Then

  $$\omega' \in \mathsf{Mod}_{\mathsf{LP}}\big(K \cup_i \{\alpha\}\big), \quad \text{and} \quad \mathrm{r}_K\big(\omega'\big) = \mathrm{r}_K(\omega).$$

  So,

  $$\min\big\{\mathrm{r}_K(\omega) | \omega \in \mathsf{Mod}_{\mathsf{LP}}(K)\big\} = \min\big\{\mathrm{r}_{K \cup_i \{\alpha\}}(\omega) \bigm| \omega \in \mathsf{Mod}_{\mathsf{LP}}\big(K \cup_i \{\alpha\}\big)\big\}.$$

  That is,

  $$\mathrm{Inc}_{\mathrm{LP_m}}(K) = \mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big).$$

- *Inferior Stratum*. Suppose that $1 \leqslant j \leqslant i \leqslant n$. Note that $\mathsf{Mod}_{\mathsf{LP}}(K \cup_i \{\alpha\}) = \mathsf{Mod}_{\mathsf{LP}}(K \cup_j \{\alpha\})$. $\forall \omega \in \mathsf{Mod}_{\mathsf{LP}}(K \cup_i \{\alpha\})$, we use $\mathrm{r}_{K \cup_i \{\alpha\}}(\omega)$ and $\mathrm{r}_{K \cup_j \{\alpha\}}(\omega)$ to denote the ranks of ω in $K \cup_i \{\alpha\}$ and $K \cup_j \{\alpha\}$, respectively.
  If $\omega(\alpha) = \mathsf{B}$, then

  $$\mathrm{r}_{K \cup_i \{\alpha\}}(\omega) < \mathrm{r}_{K \cup_j \{\alpha\}}(\omega)$$

  because

  $$\big|\big\{\gamma \in S_j \bigm| \omega(\gamma) = \mathsf{B}\big\}\big| + 1 \leqslant \big|\big\{\gamma \in S_j \cup \{\alpha\} \bigm| \omega(\gamma) = \mathsf{B}\big\}\big|.$$

  If $\omega(\alpha) = \mathsf{T}$, then

  $$\mathrm{r}_{K \cup_i \{\alpha\}}(\omega) = \mathrm{r}_{K \cup_j \{\alpha\}}(\omega).$$

  That is, for any ω,

  $$\mathrm{r}_{K \cup_i \{\alpha\}}(\omega) \leqslant \mathrm{r}_{K \cup_j \{\alpha\}}(\omega).$$

  So,

  $$\mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big) \leqslant \mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_j \{\alpha\}\big) \quad \text{if } i \geqslant j. \qquad \square$$

Evidently, we can get the following corollary from Proposition 5.4.

Corollary 5.2. *The* $\mathsf{WLP_m}$ *measure* $\mathrm{Inc}_{\mathsf{w}}$ *satisfies* Consistency *and* Safe Formula Independence.

Proof. Let $K = (S_1, S_2, \ldots, S_n)$ be a stratified knowledge base.

- *Consistency*. $\mathrm{Inc}_{\mathsf{w}}(K) = 0. \Leftrightarrow \mathrm{Inc}_{\mathrm{LP_m}}(K) = \vec{0}. \Leftrightarrow K$ is consistent.
- *Safe Formula Independence*. Suppose that $\mathrm{Var}(\{\alpha\}) \cap \mathrm{Var}(K) = \emptyset$ and $\alpha \nvdash \bot$. From Proposition 5.4,

  $$\mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big) = \mathrm{Inc}_{\mathrm{LP_m}}(K).$$

  Further,

  $$\mathrm{Inc}_{\mathrm{LP_m}}\big(K \cup_i \{\alpha\}\big) \cdot \vec{\mathsf{w}}^{\tau} = \mathrm{Inc}_{\mathrm{LP_m}}(K) \cdot \vec{\mathsf{w}}^{\tau}.$$

  That is,

  $$\mathrm{Inc}_{\mathsf{w}}\big(K \cup_i \{\alpha\}\big) = \mathrm{Inc}_{\mathsf{w}}(K). \qquad \square$$

Proposition 5.5. *The* $\mathsf{WLP_m}$ *measure* $\mathrm{Inc_w}$ *does not satisfy* Monotony, Dominance, *and* Inferior Stratum.

Proof. We only need to provide two counterexamples.

- Consider the following counterexample for *Monotony* and *Dominance*: Let $\vec{\mathrm{w}} = (w_1, w_2, w_3)$ and $K = (\{a\}, \{\neg a \vee b\}, \{\neg b \wedge c_1, \neg b \wedge c_2, \ldots, \neg b \wedge c_k\})$, where $k = \lceil \frac{2w_1}{w_3} \rceil + 2$. Then

$$\mathrm{Inc_w}\big(K \cup_1 \{\neg a\}\big) = 2w_1 + w_2 < w_2 + k \cdot w_3 = \mathrm{Inc_w}(K),$$
$$\mathrm{Inc_w}\big(K \cup_1 \{\neg a \wedge c_1\}\big) = 2w_1 + w_2 < w_2 + k \cdot w_3 = \mathrm{Inc_w}\big(K \cup_1 \{c_1\}\big).$$

- Consider the following counterexample for *Inferior Stratum*: Let $\vec{\mathrm{w}} = (w_1, w_2, w_3, w_4)$ and $K = (\{a\}, \{\neg b \vee c\}, \{\neg c\}, \{b \wedge e_1, \ldots, b \wedge e_k\})$, where $k = \lceil \frac{w_3}{w_4} \rceil + 1$. Consider

$$K \cup_1 \{b\} = \big(\{a, b\}, \{\neg b \vee c\}, \{\neg c\}, \{b \wedge e_1, \ldots, b \wedge e_k\}\big),$$
$$K \cup_4 \{b\} = \big(\{a\}, \{\neg b \vee c\}, \{\neg c\}, \{b \wedge e_1, \ldots, b \wedge e_k, b\}\big).$$

Then

$$\mathrm{Inc_{LP_m}}\big(K \cup_1 \{b\}\big) = (0, 1, 1, 0) > (0, 1, 0, k+1) = \mathrm{Inc_{LP_m}}\big(K \cup_4 \{b\}\big),$$

but

$$\mathrm{Inc_w}\big(K \cup_1 \{b\}\big) = w_2 + w_3 < w_2 + (k+1)w_4 = \mathrm{Inc_w}\big(K \cup_4 \{b\}\big). \qquad \Box$$

Compared to $\mathrm{Inc_{LP_m}}$, $\mathrm{Inc_w}$ only supports two intuitive properties, i.e., *Consistency* and *Safe Formula Independence*, and fails to support *Monotony*, *Dominance*, and *Inferior stratum*. This is largely due to weakening of the difference between preference levels of strata in integration process based on weights. In this sense, $\mathrm{Inc_{LP_m}}$ can be used to capture the nature of inconsistency for stratified knowledge bases in essence, whilst $\mathrm{Inc_w}$ is considered meaningful only in the case of a single value being more attractive.

As the normalized version of $\mathrm{Inc_{LP_m}}$, the $\mathsf{NLP_m}$ measure $\mathrm{Inc_{N_{LP_m}}}$ possesses the following two intuitive properties.

Proposition 5.6. *The* $\mathsf{NLP_m}$ *inconsistency measure* $\mathrm{Inc_{N_{LP_m}}}(K)$ *satisfies* Consistency *and* Dominance.

Proof. Let $K = (S_1, \ldots, S_n)$.

- *Consistency*. $\mathrm{Inc_{N_{LP_m}}}(K) = \vec{0}. \Leftrightarrow \mathrm{Inc_{LP_m}}(K) = \vec{0}. \Leftrightarrow K$ is consistent.
- *Dominance*. From the proof for *Dominance* in Proposition 5.4, we have known that

$$\mathrm{r}_{K \cup_i \{\beta\}}(\omega) \leqslant \mathrm{r}_{K \cup_i \{\alpha\}}(\omega).$$

Suppose that $K \cup_i \{\alpha\} = (T_1, \ldots, T_n)$ and $K \cup_i \{\beta\} = (R_1, \ldots, R_n)$. Then

$$|T_i| = |R_i| \quad \text{for all } 1 \leqslant i \leqslant n.$$

So,

$$\mathrm{r}^{\mathrm{N}}_{K \cup_i \{\beta\}}(\omega) \leqslant \mathrm{r}^{\mathrm{N}}_{K \cup_i \{\alpha\}}(\omega).$$

Therefore,

$$\mathrm{Inc_{N_{LP_m}}}\big(K \cup_i \{\alpha\}\big) \geqslant \mathrm{Inc_{N_{LP_m}}}\big(K \cup_i \{\beta\}\big). \qquad \Box$$

Moreover, $\mathrm{Inc_{N_{LP_m}}}$ does not support *Safe Formula Independence*, *Monotony* and *Inferior Stratum*, as illustrated by the following example.

Example 5.2. Consider $K = (\{a\}, \{\neg a \vee b\}, \{\neg b, c\})$ again. Suppose that we add a safe formula d to the third stratum of K, i.e., $K \cup_3 \{d\} = (\{a\}, \{\neg a \vee b\}, \{\neg b, c, d\})$. Then

$$\mathrm{Inc_{N_{LP_m}}}\big(K \cup_3 \{d\}\big) = \left(0, 1, \frac{1}{3}\right) < \left(0, 1, \frac{1}{2}\right) = \mathrm{Inc_{N_{LP_m}}}(K).$$

Note that adding d to S_3 enlarges the size of S_3. But d brings no new inconsistency. This is why the inconsistency in K is *diluted* by adding d.

Table 1
Logical properties of inconsistency measures.

Properties	Inconsistency measures				
	$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}$	$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$	$\mathrm{Inc}_{\mathrm{LP_m}}$	$\mathrm{Inc}_{\mathrm{W}}$	$\mathrm{Inc}_{\mathrm{N_{LP_m}}}$
Consistency	√	√	√	√	√
Monotony	√	√	√	×	–
Dominance	√	×	√	×	√
Safe formula independence	√	√	√	√	–
Inferior stratum	√	√	√	×	–
Free formula independence	–	√	–	–	–

Moreover,

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}\left(K \cup_3 \{d\}\right) = \left(0, 1, \frac{1}{3}\right) > \left(0, \frac{1}{2}, \frac{1}{2}\right) = \mathrm{Inc}_{\mathrm{N_{LP_m}}}\left(K \cup_2 \{d\}\right).$$

As mentioned above, none of *Safe Formula Independence*, *Monotony* and *Inferior Stratum* is designed for characterizing the normalized inconsistency measure which focuses on the proportion of inconsistency in a knowledge base. Therefore, as a normalized version, its failure in supporting *Safe Formula Independence*, *Monotony* and *Inferior Stratum* cannot be considered as a negative aspect of $\mathrm{Inc}_{\mathrm{N_{LP_m}}}$. However, how to develop properties suitable for such normalized inconsistency measures is still an open problem.

Finally, we summarize these results in Table 1, in which $\surd$ denotes satisfiable properties, $\times$ unsatisfiable ones, and $-$ unconcerned ones.

6. Computational complexity

Now we turn to the complexity issue. We assume that the reader is familiar with the basics of complexity, in particular the polynomial hierarchy ($\Sigma_0^p = \Delta_0^p = \mathrm{N}$, and for all $k \geqslant 0$, $\Sigma_{k+1}^p = \mathrm{NP}^{\Sigma_k^p}$, and $\Delta_{k+1}^p = \mathrm{P}^{\Sigma_k^p}$) [32]. Here $\mathrm{FP}^{\mathrm{NP}}$ (resp. $\mathrm{FP}^{\Sigma_2^p}$) is the class of all functions that can be computed by a polynomial-time Turing machine with an NP oracle (resp. an Σ_2^p oracle), and $\mathrm{FP}^{\mathrm{NP}[\log n]} \subseteq \mathrm{FP}^{\mathrm{NP}}$ is the class of all functions that can be computed in polynomial time by a Turing machine using a number of NP oracles bounded by a logarithmic function of the size of the input. The canonical natural $\mathrm{FP}^{\mathrm{NP}}$-complete problem is MAX-WEIGHT SAT, which is given as follows: *Given a set of clauses, each with an integer weight, find the truth assignment that satisfies a set of clauses with the most total weight* [32]. The problem of MAX SAT, one of the well known natural $\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete problems, is given as follows: *Given a set of clauses, find the truth assignment that satisfies a set of clauses with the maximal number* [32].

Let us recall the complexity result about the $\mathrm{LP_m}$ inconsistency measure $\mathrm{I}_{\mathrm{LP_m}}(K)$ in the case of flat knowledge bases. It has been shown that the problem of computing $\mathrm{I}_{\mathrm{LP_m}}(K)$ is $\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete in the case of flat knowledge bases [14]. Note that

$$\mathrm{C}_{\mathrm{LP_m}}(K) = \left|\mathrm{Var}(K)\right| \cdot \mathrm{I}_{\mathrm{LP_m}}(K),$$

then we can get the following result firstly.

Lemma 6.1. *The problem of computing* $\mathrm{C}_{\mathrm{LP_m}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}[\log n]}$*-complete in the case of flat knowledge bases.*

We omit the proof of this lemma, because the membership is obvious, and the proof for hardness has no essential difference from that for Theorem 4 in [33].

We have identified the following complexity of computing the naive MSIM $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K)$.

Proposition 6.1. *The problem of computing* $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}}$*-complete.*

Proof. Membership: Suppose that $K = (S_1, \ldots, S_n)$. As shown by Lemma 6.1, for each $1 \leqslant i \leqslant n$, $\mathrm{C}_{\mathrm{LP_m}}(K(i))$ can be computed in polynomial time by a deterministic algorithm using a number of calls to an NP oracle bounded by a logarithmic function of $(|K(i)| + |\mathrm{Var}(K(i))|)$. Clearly, $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K) = (\mathrm{C}_{\mathrm{LP_m}}(K(1)), \ldots, \mathrm{C}_{\mathrm{LP_m}}(K(n)))$ can be computed in polynomial time by a deterministic algorithm using a number of calls to an NP oracle bounded by a polynomial function of $(n + |K| + |\mathrm{Var}(K)|)$.

Hardness: We shall reduce MAX-WEIGHT SAT to it. Given any set of clauses $C_1, C_2, \ldots, C_m$ on l variables $x_1, x_2, \ldots, x_l$, with weights $w_1, w_2, \ldots, w_m$, we construct a stratified knowledge base consisting of two strata as follows: $K = (S_1, S_2)$, where

$$S_1 = \{x_i \wedge \neg x_i \to \mathbb{F} \mid i = 1, 2, \ldots, l\};$$
$$S_2 = \bigcup_{i=1}^{m} \{C_i \vee q_{i,j}, \neg q_{i,j} \mid j = 1, 2, \ldots, w_i\},$$

where $q_{i,j}$ is a new variable out of $\{x_1, x_2, \ldots, x_n\}$ for each (i, j). Note that $|S_2| = 2 \cdot \sum_{i=1}^{m} w_i$. Clearly, this reduction can be done in polynomial time w.r.t. $(m + l + \sum_{i=1}^{m} w_i)$.

Let ω be a $\mathsf{LP_m}$ interpretation over $\{x_1, x_2, \ldots, x_n, q_{1,1}, \ldots, q_{m,w_m}\}$. If ω is a $\mathsf{LP_m}$ model for S_1, then $\omega(x_i) \neq \mathsf{B}$ for all $1 \leqslant i \leqslant l$. This implies that $\omega(C_j) \neq \mathsf{B}$ as well for all $1 \leqslant j \leqslant m$. In particular, if $\omega(C_j) = \mathsf{F}$, then $\omega(q_j) = \mathsf{B}$ must hold.

Let A be a classical interpretation on $\{x_1, x_2, \ldots, x_l\}$, we define an $\mathsf{LP_m}$ interpretation ω_A over $\{x_1, x_2, \ldots, x_l, q_{1,1}, \ldots, q_{m,w_m}\}$ as follows:

$$\omega_A(x_i) = A(x_i), \quad \text{for all } 1 \leqslant i \leqslant l;$$
$$\omega_A(q_{i,j}) = \begin{cases} \mathsf{B}, & \text{if } A(C_i) = \mathsf{F}, \\ \mathsf{F}, & \text{if } A(C_i) = \mathsf{T}. \end{cases} \quad \text{for all } 1 \leqslant j \leqslant w_i.$$

Let $C_{n_1}, \ldots, C_{n_l}$ be the set of clauses satisfied by A, then

$$\left|\omega_A!\big(K(2)\big)\right| = \sum_{i=1}^{m} w_i - \sum_{j=1}^{l} w_{n_j}.$$

Then A is an interpretation that satisfies the set of clause above with the most total weight if and only if ω_A is a minimal model of $K(2)$, moreover,

$$\mathrm{Inc}_{\mathrm{m}}^{\mathsf{C_{LP}}}(K) = \big(0, \left|\omega_A!\big(K(2)\big)\right|\big). \qquad \square$$

The following proposition shows that fixing the number of strata can lower the complexity of computing $\mathrm{Inc}_{\mathrm{m}}^{\mathsf{C_{LP}}}(K)$.

Proposition 6.2. *If the number n of strata is constant, the problem of computing* $\mathrm{Inc}_{\mathrm{m}}^{\mathsf{C_{LP}}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}[\log n]}$*-complete.*

Proof. Membership: Let $K = (S_1, \ldots, S_n)$. As shown by Lemma 6.1, for each $1 \leqslant i \leqslant n$, $\mathsf{C_{LP_m}}(K(i))$ can be computed in polynomial time by a deterministic algorithm using a number of calls to an NP oracle bounded by $O(\log(|K(i)| + |\mathrm{Var}(K(i))|))$. Note that $|K(i)| \leqslant |K|$ and $|\mathrm{Var}(K(i))| \leqslant |\mathrm{Var}(K)|$, then $\mathrm{Inc}_{\mathrm{m}}^{\mathsf{C_{LP}}}(K) = (\mathsf{C_{LP_m}}(K(1)), \ldots, \mathsf{C_{LP_m}}(K(n)))$ can be computed in polynomial time by a deterministic algorithm using a number of calls to an NP oracle bounded by $O(n \log(|K| + |\mathrm{Var}(K)|))$. Allowing for n being a constant, then the problem is in $\mathrm{FP}^{\mathrm{NP}[\log n]}$.

Hardness: it can be proved by the following reduction from MAX SAT. Given any set of clauses $C_1, C_2, \ldots, C_m$ on l variables $x_1, x_2, \ldots, x_l$, we construct a stratified knowledge base with n-stratum as follows:

$$K = (S_1, S_2, \ldots, S_n),$$

where

$$S_i = \{p_k \wedge \neg p_k \to \mathbb{F}\} \quad \text{for all } 1 \leqslant k \leqslant n - 1;$$
$$S_n = \{x_i \wedge \neg x_i \to \mathbb{F} \mid 1 \leqslant i \leqslant l\} \cup \{C_j \vee q_j, \neg q_j \mid 1 \leqslant j \leqslant m\}.$$

Here p_k and q_j are new variables out of $\{x_1, \ldots, x_l\}$ for all k and j. Clearly, this reduction can be done in polynomial time w.r.t. $(m + l)$.

Let A be a classical interpretation on $\{x_1, x_2, \ldots, x_l\}$, we define an $\mathsf{LP_m}$ interpretation ω_A over $\{x_1, x_2, \ldots, x_l, p_1, \ldots, p_{n-1}, q_1, \ldots, q_m\}$ as follows:

$$\omega_A(x_i) = A(x_i), \quad \text{for all } 1 \leqslant i \leqslant l;$$
$$\omega_A(p_i) = \mathsf{T}, \quad \text{for all } 1 \leqslant i \leqslant n - 1;$$
$$\omega_A(q_j) = \begin{cases} \mathsf{B}, & \text{if } A(C_j) = \mathsf{F}, \\ \mathsf{F}, & \text{if } A(C_j) = \mathsf{T}. \end{cases}$$

Let k be the number of clauses satisfied by A, then

$$\left|\omega_A!\big(K(n)\big)\right| = m - k.$$

Then A is an interpretation that satisfies the subset of clauses with the maximal number if and only if ω_A is a minimal model of K, moreover,

$$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{C_{LP}}}(K) = \big(0, \ldots, 0, \big|\omega_A!\big(K(n)\big)\big|\big). \qquad \square$$

The stratum-centric MSIM $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$ is a multi-section inconsistency measure built upon minimal inconsistent subsets. The main part of computing $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K)$ is essentially checking whether a formula is involved in minimal inconsistent subsets due to each stratum. It has been shown that checking whether a set of clauses is a minimal inconsistent subset (also termed Minimally unsatisfiable subformulas or MUS) or not is DP-complete, and checking whether a formula is involved in minimal inconsistent subsets of a knowledge base is in Σ_2^p [34]. We may obtain the following lower complexity bound for the stratum-centric inconsistency measure. Its tight complexity bound is still open.

Proposition 6.3. *The problem of computing* $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K)$ *is in* $\mathrm{FP}^{\Sigma_2^p}$.

Proof. Let $K = (S_1, \ldots, S_n)$. Suppose that $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K) = (c_1, \ldots, c_n)$ and $T_i = R_i = \emptyset$ for all $1 \leqslant i \leqslant n$. consider the following algorithm:

(1) if $S_1 \vdash \bot$, then $c_1 = 1$; else $c_1 = 0$;
(2) for each $i \leqslant 2$, if $K(i) \nvdash \bot$, then $c_i = 0$; else checking whether α is involved in minimal inconsistent subsets of $K(i)$ for each $\alpha \in S_i$, if yes, $T_i = T_i \cup \{\alpha\}$.
(3) if $T_i = \emptyset$, then $c_i = 0$, else go to (4);
(4) for k from 1 to $i-1$, identifying the set R_k of formulas in S_i involved in minimal inconsistent subsets of $K(i) \setminus S_k$, if $R_k \subset T_i$, then $c_i = i + 1 - k$, break; else $k = k + 1$.
(5) if $k = i$, then $c_i = 1$.

Note that both R_k and T_i can be computed by using $|S_i|$ calls to a Σ_2^p oracle. This means c_i can be computed by using a number of calls to a Σ_2^p oracle bounded by $O(i|S_i|)$. Therefore, $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K)$ can be computed by using a number of calls to a Σ_2^p oracle bounded by $O(|K|^2)$. So, the problem of computing $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K)$ is in $\mathrm{FP}^{\Sigma_2^p}$. $\square$

Concerning the $\mathsf{PLP_m}$ measure $\mathrm{Inc}_{\mathrm{LP_m}}$, we have obtained the following complexity results.

Proposition 6.4. *The problem of computing* $\mathrm{Inc}_{\mathrm{LP_m}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}}$*-complete.*

Proof. Given a stratified knowledge base $K = (S_1, S_2, \ldots, S_n)$, note that given a $\mathsf{LP_m}$ interpretation ω,

(1) deciding whether ω is a $\mathsf{LP_m}$ model of K is polynomial w.r.t. $(|\mathrm{Var}(K)| + |K|)$;
(2) if ω is a $\mathsf{LP_m}$ model of K, computing $\mathrm{r_K}(\omega) = (r_1, r_2, \ldots, r_n)$ is polynomial w.r.t. $(|\mathrm{Var}(K)| + |K|)$.

Suppose that $\mathrm{Inc}_{\mathrm{LP_m}}(K) = (c_1, c_2, \ldots, c_n)$. Then we consider the following nondeterministic algorithm for deciding whether $c_i \leqslant b$ given b:

- guess an interpretation ω;
- check that ω is a model of K;
- if so, compute $\mathrm{r_K}(\omega) = (r_1, r_2, \ldots, r_n)$ and check that $r_i \leqslant b$.

This algorithm shows that deciding whether $c_i \leqslant b$ is in NP.

Further, $c_1, c_2, \ldots, c_n$ can be determined in the following order, where c_i is computed by queries to the NP oracle above whether there exists a $\mathsf{LP_m}$ model ω such that $r_j = c_j$, for $1 \leqslant j < i$, and $r_i \leqslant b$ for each i, where b is a given number. This can be done with a number of NP oracle calls bounded by a polynomial function of $(n + |\mathrm{Var}(K)| + |K|)$. So, this problem is in $\mathrm{FP}^{\mathrm{NP}}$.

To prove completeness, we shall reduce MAX-WEIGHT SAT to it. Given any set of clauses $C_1, C_2, \ldots, C_m$ on l variables $x_1, x_2, \ldots, x_l$, with weights $w_1, w_2, \ldots, w_m$, we construct a stratified knowledge base consisting of two strata as follows: $K = (S_1, S_2)$, where

$$S_1 = \{x_i \vee \neg x_i \mid i = 1, 2, \ldots, l\};$$
$$S_2 = \bigcup_{i=1}^{m} \{C_i \vee q_{i,j}, \neg q_{i,j} \mid j = 1, 2, \ldots, w_i\},$$

where $q_{i,j}$ is a new variable out of $\{x_1, x_2, \ldots, x_n\}$. Note that $|S_2| = 2 \cdot \sum_{i=1}^{m} w_i$. Clearly, this reduction can be done in polynomial time w.r.t. $(m + l + \sum_{i=1}^{m} w_i)$.

Let ω be a $\mathsf{LP_m}$ interpretation over $\{x_1, x_2, \ldots, x_n, q_{1,1}, \ldots, q_{m,w_m}\}$, and A a classical interpretation over $\{x_1, x_2, \ldots, x_n\}$. ω is compatible with A if $\omega(x_i) = A(x_i)$ for all $1 \leqslant i \leqslant n$.

Note that for any $\mathsf{LP_m}$ model ω compatible with a classical interpretation A, then $\omega(x_i \vee \neg x_i) = \mathsf{T}$, So $\mathrm{r_K}(\omega) = (0, k)$, where k is the number of formulas in S_2 assigned B under ω. So, if $\mathrm{Inc_{LP}}(K) = \mathrm{r_K}(\omega)$, then there exists a classical interpretation A compatible with ω. That is, under ω, $\omega(C_i) \in \{\mathsf{T}, \mathsf{F}\}$.

In particular, if $\omega(C_i) = \mathsf{F}$, then $\omega(q_{i,j}) = \mathsf{B}$, and

$$\omega(C_i \vee q_{i,j}) = \omega(\neg q_{i,j}) = \mathsf{B}.$$

Otherwise, $\omega(q_{i,j}) \in \{\mathsf{B}, \mathsf{F}\}$. To minimize the number of inconsistent formulas, $\omega(q_{i,j}) = \mathsf{F}$. Because

$$\omega(C_i \vee q_{i,j}) = \omega(\neg q_{i,j}) = \mathsf{T}.$$

if $\omega(q_{i,j}) = \mathsf{F}$.

Let A be a classical interpretation on $\{x_1, x_2, \ldots, x_n\}$, we define a $\mathsf{LP_m}$ interpretation ω_A over $\{x_1, x_2, \ldots, x_n, q_{1,1}, \ldots, q_{m,w_m}\}$ as follows:

$$\omega_A(x_i) = A(x_i), \quad \text{for all } 1 \leqslant i \leqslant n;$$

$$\omega_A(q_{i,j}) = \begin{cases} \mathsf{B}, & \text{if } A(C_i) = \mathsf{F}, \\ \mathsf{F}, & \text{if } A(C_i) = \mathsf{T}. \end{cases} \quad \text{for all } 1 \leqslant j \leqslant w_i.$$

Let $C_{n_1}, \ldots, C_{n_l}$ be the set of clauses satisfied by A, then $\mathrm{r_K}(\omega_A) = (0, 2 \cdot (\sum_{i=1}^{m} w_i - \sum_{j=1}^{l} w_{n_j}))$.

So, A is an interpretation that satisfies the set of clauses above with the most total weight if and only if $\mathrm{Inc_{LP_m}}(K) = \mathrm{r_K}(\omega_A)$. $\square$

Further, we have obtained that fixing the number of strata can lower the upper bound of complexity of computing $\mathrm{Inc_{LP_m(K)}}$.

Proposition 6.5. *If the number n of strata is constant, then the problem of computing* $\mathrm{Inc_{LP_m}}(K)$ *is* $\mathrm{FP^{NP[\log n]}}$*-complete.*

Proof. Membership: From Proposition 6.4, we know computing $\mathrm{Inc_{LP_m}}(K)$ can be done with polynomially many NP oracle calls.

Note that $0 \leqslant c_i \leqslant |S_i|$ for all $1 \leqslant i \leqslant n$. So, c_i can be computed by using a binary search on $\{0, \ldots, |S_i|\}$ with a call to an NP oracle to check whether $r_i \leqslant b$. Since a binary search on $\{0, \ldots, |S_i|\}$ needs at most $O(\log_2 |S_i|)$ steps, then $r_i = c_i$ can be computed using $O(\log_2 |S_i|)$ calls to an NP oracle.

Let $|K| = \sum_{i=1}^{n} |S_i|$. Then computing $c_1, c_2, \ldots, c_n$ can be done with $O(n \log_2 |K|)$ many NP oracle calls. If n is constant, then computing $\mathrm{Inc_{LP_m}}(K)$ is in $\mathrm{FP^{NP[\log n]}}$.

Hardness: it can be proved by the following reduction from MAX SAT. Given any set of clauses $C_1, C_2, \ldots, C_m$ on l variables $x_1, x_2, \ldots, x_l$, we construct a stratified knowledge base with n-stratum as follows:

$$K = (S_1, S_2, \ldots, S_n),$$

where

$$S_i = \{p_k \wedge \neg p_k \to \mathbb{F}\} \quad \text{for all } 1 \leqslant k \leqslant n-1;$$

$$S_n = \{x_i \wedge \neg x_i \to \mathbb{F} \mid 1 \leqslant i \leqslant l\} \cup \{C_j \vee q_j, \neg q_j \mid 1 \leqslant j \leqslant m\}.$$

Here p_k and q_j are new variables out of $\{x_1, \ldots, x_l\}$ for all k and j. Clearly, this reduction can be done in polynomial time w.r.t. $(m + l)$.

Let A be a classical interpretation on $\{x_1, x_2, \ldots, x_l\}$, we define an $\mathsf{LP_m}$ interpretation ω_A over $\{x_1, x_2, \ldots, x_l, p_1, \ldots, p_{n-1}, q_1, \ldots, q_m\}$ as follows:

$$\omega_A(x_i) = A(x_i), \quad \text{for all } 1 \leqslant i \leqslant l;$$

$$\omega_A(p_i) = \mathsf{T}, \quad \text{for all } 1 \leqslant i \leqslant n-1;$$

$$\omega_A(q_j) = \begin{cases} \mathsf{B}, & \text{if } A(C_j) = \mathsf{F}, \\ \mathsf{F}, & \text{if } A(C_j) = \mathsf{T}. \end{cases}$$

Let k be the number of clauses satisfied by A, then

$$\left|\omega_A!\big(K(n)\big)\right| = m - k.$$

Then A is an interpretation that satisfies the subset of clauses with the maximal number if and only if ω_A is a minimal model of K, moreover,

$$\mathrm{Inc}_{\mathrm{LP_m}}(K) = \bigl(0, \ldots, 0, 2\bigl|\omega_A!\bigl(K(n)\bigr)\bigr|\bigr). \qquad \square$$

Clearly, we can obtain the following complexity results from Propositions 6.4 and 6.5.

Corollary 6.1. *The problem of computing* $\mathrm{Inc}_{\mathrm{w}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}}$*-complete.*

Proof. Membership: Note that $\mathrm{Inc}_{\mathrm{w}}(K) = \mathrm{Inc}_{\mathrm{LP_m}}(K) \cdot \vec{\mathrm{w}}^{\tau}$. Then the problem of computing $\mathrm{Inc}_{\mathrm{w}}(K)$ is in $\mathrm{FP}^{\mathrm{NP}}$.

Hardness: we construct the same reduction as that in the proof of Proposition 6.4. Then A is an interpretation that satisfies the set of clauses with the most total weight if and only if $\mathrm{Inc}_{\mathrm{w}}(K) = \mathrm{r_K}(\omega_A) \cdot \vec{\mathrm{w}}^{\tau}$. $\square$

Corollary 6.2. *If the number n of strata is a constant, the problem of computing* $\mathrm{Inc}_{\mathrm{w}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}[\log n]}$*-complete.*

Proof. From Proposition 6.5, Membership is clear.

Hardness: we construct the same reduction as that in the proof of Proposition 6.5. Then A is an interpretation that satisfies the subset of clauses with the maximal number if and only if ω_A is a minimal model of K, moreover,

$$\mathrm{Inc}_{\mathrm{w}}(K) = \bigl(0, \ldots, 0, 2\bigl|\omega_A!\bigl(K(n)\bigr)\bigr|\bigr) \cdot \vec{\mathrm{w}}^{\tau}. \qquad \square$$

For the $\mathrm{NLP_m}$ measure $\mathrm{Inc}_{\mathrm{N_{LP_m}}}$, we identified the following complexity results.

Proposition 6.6. *The problem of computing* $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}}$*-complete.*

Proof. Membership is clear if we replace $\mathrm{r_K}(\omega)$ with $\mathrm{r_K^N}(\omega)$ in the proof of Proposition 6.4.

Hardness: We construct the same reduction as that used in the proof of Proposition 6.4. So, A is an interpretation that satisfies the set of clauses with the most total weight if and only if

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K) = \mathrm{r_K}(\omega_A) \cdot \begin{bmatrix} \frac{1}{|S_1|} & 0 \\ 0 & \frac{1}{|S_2|} \end{bmatrix}. \qquad \square$$

Proposition 6.7. *If the number n of strata is a constant, the problem of computing* $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ *is* $\mathrm{FP}^{\mathrm{NP}[\log n]}$*-complete.*

Proof. Membership: From Proposition 6.6, we know computing $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ can be done with polynomially many NP oracle calls.

Note that $0 \leqslant c_i \leqslant 1$ for all $1 \leqslant i \leqslant n$. So, c_i can be computed by using a binary search on $\{0, \frac{1}{|S_i|}, \ldots, \frac{|S_{i-1}|}{|S_i|}, 1\}$ with a call to an NP oracle to check whether $r_i \leqslant b$. Since a binary search on $\{0, \frac{1}{|S_i|}, \ldots, \frac{|S_{i-1}|}{|S_i|}, 1\}$ needs at most $O(\log_2 |S_i|)$ steps, then $r_i = c_i$ can be computed using $O(\log_2 |S_i|)$ calls to an NP oracle.

Let $|K| = \sum_{i=1}^{n} |S_i|$. Then computing $c_1, c_2, \ldots, c_n$ can be done with $O(n \log_2(|K| + |Var(K)|))$ many NP oracle calls. If n is constant, then computing $\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K)$ is in $\mathrm{FP}^{\mathrm{NP}[\log n]}$.

Hardness: we construct the same reduction as that in the proof of Proposition 6.5. Then A is an interpretation that satisfies the subset of clauses with the maximal number if and only if ω_A is a minimal model of K, moreover,

$$\mathrm{Inc}_{\mathrm{N_{LP_m}}}(K) = \left(0, \ldots, 0, \frac{2|\omega_A!(K(n))|}{2m + l}\right). \qquad \square$$

In summary, the complexity of each inconsistency measure built upon $\mathrm{LP_m}$ models presented in this paper is $\mathrm{FP}^{\mathrm{NP}}$-complete. Moreover, when we fix the number n of strata, its complexity reduce to $\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete.

Finally, we summarize the complexity results about these measures in Table 2.

7. An application in requirements engineering

Here we use a small but explanatory example in requirements engineering to illustrate the application of our measures for stratified knowledge base. We consider a scenario for eliciting requirements for updating an existing software used in [18]. Moreover, to illustrate the role of our measures in monitoring inconsistency handling process, we adapt slightly the scenario to the case of prioritized requirements by giving a preference relation over requirement statements in the scenario.

Table 2
Computational complexity.

Measures	Computational complexity	
	General	Fixing the number n of strata
$\mathrm{Inc}_{\mathrm{m}}^{C_{\mathrm{LP}}}$	$\mathrm{FP}^{\mathrm{NP}}$-complete	$\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete
$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$	$\mathrm{FP}^{\Sigma_2^p}$	$\mathrm{FP}^{\Sigma_2^p}$
$\mathrm{Inc}_{\mathrm{LP_m}}$	$\mathrm{FP}^{\mathrm{NP}}$-complete	$\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete
$\mathrm{Inc}_{\mathrm{N_{LP_m}}}$	$\mathrm{FP}^{\mathrm{NP}}$-complete	$\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete
$\mathrm{Inc}_{\mathrm{W}}$	$\mathrm{FP}^{\mathrm{NP}}$-complete	$\mathrm{FP}^{\mathrm{NP}[\log n]}$-complete

Example 7.1. Consider the following scenario for eliciting requirements for updating an existing software. There are three stakeholders involved in this scenario, including the seller of the new system, the user of the existing system (the user for short), and the domain expert in requirements engineering. Each of the three stakeholders may provide demands from her/his own perspective. When inconsistencies in their demands are identified, developers and the three stakeholders start to negotiate on resolving inconsistencies. Inconsistency measures could be used to assess the progress of each round of negotiation in inconsistency handling.

- The seller of the new system provides two demands:
  (a1) The user interface of the system-to-be should be in the modern idiom (i.e., fashionable).
  (a2) The system-to-be should be developed based on the newest development techniques.
- The user of the existing system provides three demands:
  (b1) The system-to-be should be developed based on the techniques used in the existing system.
  (b2) The user interface of the system-to-be should maintain the style of the existing system.
  (b3) The system-to-be should be secure.
- The domain expert in requirements engineering provides one constraint about security:
  (c1) To guarantee the security of the system-to-be, openness (or ease of extension) should not be considered.

The security is considered as one of the most important requirements for a system, so *(c1)* and *(b3)* are more important than others. Moreover, the demands of the user of the existing system are considered more important than the seller's demands. Then the set of requirements can be stratified into 3 strata, i.e.,

$$\big(\{(b_3),(c_1)\},\{(b_1),(b_2)\},\{(a_1),(a_2)\}\big).$$

The following predicates are used in [18] to formulate the requirements:

- the predicate Fash(int_f) is used to denote that the interface is fashionable;
- the predicate Open(sys) is used to denote that the system is open;
- the predicate New(sys) is used to denote that the system will be developed based on the newest techniques;
- the predicate Secu(sys) is used to denote that the system is secure.

Then the requirements above can be represented by a stratified knowledge base

$$K_R=(S_1,S_2,S_3),$$

where

$$S_1=\big\{\mathrm{Secu(sys)},\ \mathrm{Secu(sys)}\rightarrow\neg\,\mathrm{Open(sys)}\big\},$$
$$S_2=\big\{\neg\,\mathrm{New(sys)},\ \neg\,\mathrm{Fash(int\_f)}\big\},$$
$$S_3=\big\{\mathrm{Fash(int\_f)},\ \mathrm{New(sys)}\big\}.$$

Evidently, K_R is inconsistent:

$$K_R\vdash\neg\,\mathrm{Fash(int\_f)}\wedge\mathrm{Fash(int\_f)},\qquad K_R\vdash\mathrm{New(sys)}\wedge\neg\,\mathrm{New(sys)}.$$

To monitor the process of negotiation on resolving inconsistency, developers need to know how inconsistencies change after each round of negotiation. We consider using two inconsistency measures together to capture the inconsistency in requirements after each round. In detail, the stratum-centric MSIM $\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}$ will be used to capture the most preferred level of requirements involved in inconsistencies at each stratum, whilst the preference-based measure $\mathrm{Inc}_{\mathrm{LP_m}}$ is used to capture to capture how inconsistent the whole requirements is.

First, we establish the inconsistency assessment for the original requirements:

$$\mathrm{Inc}_{\mathrm{m}}^{\mathrm{s}}(K_R)=(0,0,2),\qquad \mathrm{Inc}_{\mathrm{LP_m}}(K_R)=(0,2,2).$$

The first two zero components of $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K_R)$ imply that there is no conflict between the domain expert and the user. The last component 2 in $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K_R)$ implies that the seller's demands contradict that of the user (i.e., more important demands), but do not contradict the domain expert's demands (i.e., the most important demands). On the other hand, $\mathrm{Inc}_{\mathrm{LP_m}}(K_R)$ implies that all the demands of the seller and the user are involved in conflicts between them.

Suppose that after the first round of negotiation, the seller agrees to replace requirement (a1) with a new demand:

(a3) The system-to-be should be open, that is, the system-to-be could be extended easily.

Then the revised requirements after negotiation can be represented by the following stratified knowledge base,

$$K_R^1 = \left(S_1^1, S_2^1, S_3^1\right),$$

where

$$S_1^1 = S_1, \qquad S_2^1 = S_2, \qquad S_3^1 = \left\{\mathrm{Open(sys)},\ \mathrm{New(sys)}\right\}.$$

Now the developers would like to know whether this modification may alleviate the inconsistencies in the originals. Then two inconsistency measures for K_R^1 are given as follows:

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\left(K_R^1\right) = (0,0,3) > (0,0,2) = \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K_R),$$
$$\mathrm{Inc}_{\mathrm{LP_m}}\left(K_R^1\right) = (1,1,2) > (0,2,2) = \mathrm{Inc}_{\mathrm{LP_m}}(K_R).$$

Contrary to developers expectations, this implies that the modification above does strengthen the inconsistencies in the originals. The last component 3 in $\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K_R^1)$ means that the modification brings some new conflicts between demands of the seller and that of the domain expert, whilst the first component 1 of $\mathrm{Inc}_{\mathrm{LP_m}}(K_R^1)$ implies that one of the most important demands is also involved in the conflicts.

Suppose that developers proceed with a second negotiation, and the seller agrees to abandon requirement *(a3)* because it contradicts the most important requirements, then the revised requirements after this negotiation can be represented by the following stratified knowledge base,

$$K_R^2 = \left(S_1^2, S_2^2, S_3^2\right),$$

where

$$S_1^2 = S_1, \qquad S_2^2 = S_2, \qquad S_3^2 = \left\{\mathrm{New(sys)}\right\}.$$

Now the inconsistency measures are given as follows:

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\left(K_R^2\right) = (0,0,2) = \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}(K_R) < \mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\left(K_R^1\right),$$
$$\mathrm{Inc}_{\mathrm{LP_m}}\left(K_R^2\right) = (0,1,1) < (0,2,2) = \mathrm{Inc}_{\mathrm{LP_m}}(K_R) < \mathrm{Inc}_{\mathrm{LP_m}}\left(K_R^1\right).$$

This implies that the modification above alleviate the inconsistencies in K_R^1. Now there are only conflicts between demands of the seller and that of the user in K_R^2. Moreover, compared to the originals, $\mathrm{Inc}_{\mathrm{LP_m}}(K_R^2)$ implies that only one demand of the user is involved in conflicts between them.

Suppose that after the third round of negotiation, the user agrees to withdraw requirement (b1), then the revised requirements after this negotiation can be represented by the following stratified knowledge base,

$$K_R^3 = \left(S_1^3, S_2^3, S_3^3\right),$$

where

$$S_1^3 = S_1, \qquad S_2^3 = \left\{\neg\,\mathrm{Fash(int\_f)}\right\}, \qquad S_3^2 = \left\{\mathrm{New(sys)}\right\}.$$

Now the inconsistency measures are given as follows:

$$\mathrm{Inc}^{\mathrm{s}}_{\mathrm{m}}\left(K_R^3\right) = (0,0,0), \qquad \mathrm{Inc}_{\mathrm{LP_m}}\left(K_R^3\right) = (0,0,0).$$

This means the inconsistency in the original requirements has been resolved by the modification above.

8. Related work

Inconsistency handling for stratified knowledge bases has been paid much attention in many applications. In this section, we compare the inconsistency measures presented in this paper with some of closely related approaches.

Measuring inconsistency has been increasingly considered crucial for effectively resolving inconsistency in some applications [1,16,13,17,5,6]. Most of proposals presented so far are mainly concerned with measuring inconsistency in flat knowledge bases. Note that any flat knowledge base does not convey any non-trivial preorder relation on the base explicitly. Then inconsistency measures for a flat knowledge base may be built upon either the minimum number of formulas involved in inconsistency or the minimum number of variables assigned to inconsistent truth values in some paraconsistent models [1,13]. In contrast, for the case of stratified knowledge bases, both the minimal number of formulas involved in inconsistency and the minimum number of inconsistent variables are insufficient to capture the inconsistency in a knowledge base, because the impact of preorder relation over the base on the inconsistency assessment cannot been reflected. We presented two kinds of measures to measuring inconsistency for stratified knowledge bases, i.e., the multi-section inconsistency measure and the preference-based inconsistency measure.

The multi-section inconsistency measure provides a framework to capture inconsistency in a stratified knowledge bases by using inconsistency assessments for all sections of the base together, in which inconsistency assessments for sections are defined in an incremental way. Such an inconsistency measure is more appropriate for looking inside an inconsistent stratified knowledge base stratum by stratum. We may take advantage of some inconsistency measures for flat knowledge bases such as $\mathsf{C}_{\mathsf{LP}_\mathsf{m}}$ presented in [1] to assess the inconsistency in each cut of a stratified knowledge base in the framework of multi-section inconsistency measure. We call such an instantiation naive MSIM. Note that any naive MSIM will reduce to the corresponding measures for flat knowledge base when the stratified knowledge base has only one stratum. In addition, the inconsistency rank presented in [35,36] captures the most preferred stratum where an inconsistency occurs. It can be also considered as a sketchy naive MSIM (i.e., the 0–1 vector) in which only the first i consistent cuts are assigned 0. The second instance of multi-section inconsistency measure, termed the stratum-centrical MSIM $\mathrm{Inc}^{s}_{\mathrm{m}}$, focuses on the most preferred stratum involved in new inconsistencies in each i-section due to S_i, in which inconsistencies are characterized in the form of minimal inconsistent subsets of each i-cut. It reflects the impact of preorder relation over a base on the inconsistency assessments explicitly. Note that $\mathrm{Inc}^{s}_{\mathrm{m}}$ may be reduced as the drastic inconsistency measure for flat knowledge bases when the stratified knowledge base has only one stratum, i.e., the it only tells the base is inconsistent ($\mathrm{Inc}^{s}_{\mathrm{m}}(K) = (1)$) or not ($\mathrm{Inc}^{s}_{\mathrm{m}}(K) = (0)$). Generally, the naive MSIM is more appropriate for describing how inconsistent each cut of a knowledge base is, whilst the stratum-centrical MSIM is more suitable for uncovering how preferred the strata involved in inconsistency are at each stratum.

Compared to the multi-section inconsistency measure, the preference-based inconsistency measure $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ presented in this paper aims to assess the inconsistency in a stratified knowledge base from a global or integrated perspective directly. Note that both $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ and $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$ presented in [1] are built upon LP_m models. However, the following aspects distinguish $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ and $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$. First of all, $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ is a *formula-level* inconsistency measure, whilst $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$ is a *variable-level* inconsistency measure. Roughly speaking, $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$ for a flat knowledge base focuses on the minimum number of variables assigned to the inconsistent truth value in LP_m models of the base. In contrast, $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ for a stratified knowledge base is concentrated on the number of formulas of each stratum assigned to the inconsistent truth value in LP_m models of the base, because the preorder relation is given over formulas rather than variables. Second, a $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$ model with minimal number of inconsistent truth values is not necessarily minimal with regard to the lexicographical rank of models used in $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$, and the converse does also hold. To illustrate this, consider $K = (\{a\}, \{\neg a \vee b, \neg a \vee c\}, \{\neg b, \neg c\})$. Then $\mathsf{MinMod}(K) = \{\omega_1\}$, where $\omega_1(a) = \mathsf{B}, \omega_1(b) = \omega_1(c) = \mathsf{F}$. However, $\mathrm{r}(\omega_1) = (1, 2, 0) > (0, 2, 2) = \mathrm{r}(\omega_2)$, where the model ω_2 such that $\omega_1(a) = \mathsf{T}, \omega_2(b) = \omega_2(c) = \mathsf{B}$. This implies that the two measures focus on different classes of LP_m models. Third, $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ cannot be reduced to $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}$ in the case that a stratified knowledge base has only one stratum. Lastly, $\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}$ and its normalized version are syntax sensitive. To illustrate this, consider $K = (\{a\}, \{b, c\}, \{\neg c, \neg b, d\})$ and $K' = (\{a\}, \{b\}, \{\neg c \wedge \neg b \wedge d\})$. Then $\mathsf{I}_{\mathsf{LP}_\mathsf{m}}(K(3)) = \mathsf{I}_{\mathsf{LP}_\mathsf{m}}(K'(3)) = \frac{1}{2}$, but

$$\mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}(K) = (0, 2, 2) \neq (0, 2, 1) = \mathrm{Inc}_{\mathsf{LP}_\mathsf{m}}\left(K'\right), \quad \text{and} \quad \mathrm{Inc}_{\mathsf{N}_{\mathsf{LP}_\mathsf{m}}}(K) = \left(0, 1, \frac{2}{3}\right) \neq (0, 1, 1) = \mathrm{Inc}_{\mathsf{N}_{\mathsf{LP}_\mathsf{m}}}\left(K'\right).$$

The rank of LP_m models is also different from the lexicographical rank of classical models presented in [29]. To illustrate this, consider $K = (\{a\}, \{\neg a \vee b\}, \{\neg b\})$. Consider the classical possible worlds $\{\omega_1, \omega_2, \omega_3, \omega_4\}$ where

	a	b
ω_1	T	T
ω_2	F	T
ω_3	T	F
ω_4	F	F

Let $\mathrm{l}(\omega_i) = (l_1, l_2, l_3)$ such that $l_k = |\{\alpha \in S_i | \omega(\alpha) = \mathsf{F}\}|$. Then

$$\mathrm{l}(\omega_1) = (0, 0, 1), \qquad \mathrm{l}(\omega_2) = (1, 0, 1), \qquad \mathrm{l}(\omega_3) = (0, 1, 0), \qquad \mathrm{l}(\omega_4) = (1, 0, 0).$$

Note that

$$\min\{l(\omega_1), l(\omega_2), l(\omega_3), l(\omega_4)\} = (0,0,1) \neq (0,1,1) = \min\{r(\omega) \mid \omega \in \mathsf{Mod}_{\mathsf{LP}}(K)\}.$$

The measures for prioritized knowledge bases presented in [37] is also a similar research to our approaches. At first, there is a slight difference between prioritized knowledge bases presented in [37] and stratified knowledge bases. Informally speaking, for a prioritized knowledge base, each stratum corresponds to a fixed level of preference. For example, $K = (\{a\}, \{b\}, \{\}, \{c\})$ is a prioritized knowledge base under 4-level preference scale, in which there is no formula with the third level. In contrast, each stratum in a stratified knowledge base does not necessarily corresponds to a fixed level of preference. It just expresses the relative preference of the formula in the stratum within the knowledge base. So, $(\{a\}, \{b\}, \{\}, \{c\})$ and $(\{a\}, \{\}, \{b\}, \{c\})$ are considered as different prioritized knowledge bases, but the same stratified knowledge base as $(\{a\}, \{b\}, \{c\})$. Second, the measures presented in [37] are built upon minimal inconsistent subsets of a knowledge base as well as the preference level of formulas involved in inconsistency. Roughly speaking, these inconsistency measures aims to assess inconsistency in a prioritized knowledge base by accumulating the amounts of inconsistency in all minimal inconsistent subsets (also considered as prioritized knowledge bases) of a knowledge base. In contrast, most of inconsistency measures presented in this paper are built upon LP_m models, except the stratum-centrical MSIM Inc^s_m. However, the stratum-centrical MSIM Inc^s_m is more concerned with the level of the most preferred stratum involved in minimal inconsistent subsets rather than the amount of inconsistency in each minimal inconsistent subsets.

Lastly, both the inconsistency measure $\mathrm{Inc}_\mathrm{L}(K)$ presented in [8] and the preference-based measure $\mathrm{Inc}_{\mathrm{LP_m}}$ presented in this paper are built upon LP_m models as well as the preference relation over a knowledge base. However, the main difference between $\mathrm{Inc}_\mathrm{L}(K)$ and $\mathrm{Inc}_{\mathrm{LP_m}}$ is that the latter aims to count inconsistent formulas at each stratum instead of inconsistent variables or atoms. It makes the latter more intuitive to capture the inconsistency in a stratified knowledge base because the preorder relation is explicitly given on set of formulas rather than on the set of variables.

9. Conclusion

We have presented two approaches to measuring inconsistency for stratified knowledge bases. Both the two approaches allow us to consider the impact of the total preorder relation over stratified knowledge bases on the inconsistency assessment. This paper presented the following contributions to measuring inconsistency for knowledge bases:

- We proposed the multi-section inconsistency measure for stratified knowledge bases, which focuses on inconsistency assessments for all sections of a stratified knowledge base as well as the inconsistency assessment for the stratified knowledge base. By this, we can articulate inconsistencies we meet at each stratum of the stratified knowledge base.
- We presented two kinds of stances for the multi-section inconsistency measure, i.e., the naive MSIM and the stratum-centrical MSIM. Naive MSIMs consider all the inconsistency assessments for each cut of a section of a knowledge base together as an inconsistency measure for the section. In contrast, the latter uses the level of the most preferred stratum involved in inconsistency due to each stratum of a stratified knowledge base together to capture the inconsistency in the base.
- We proposed the preference-based inconsistency measure for stratified knowledge bases, which allows us to articulate inconsistencies in a stratified knowledge base from a global perspective.
- We adapted some intuitive properties of inconsistency measures for flat bases to the case of stratified knowledge bases. In particular, we proposed a new property termed *Inferior Stratum*. We also showed that these measures defined in this paper satisfy the expected properties.
- We identified the complexity results for these measures.
- We presented a requirements engineering application to illustrate the practical potential usage of our measures.

Acknowledgements

The authors are grateful to anonymous reviewers for their valuable comments. This work was partly supported by the National Natural Science Foundation of China under Grant No. 61170300, the National Basic Research 973 program of China under Grant No. 2009CB320701, Australian Research Council (ARC) under Grant DP110101042, and the Key Project of National Natural Science Foundation of China under Grant No. 90818026.

References

[1] A. Hunter, S. Konieczny, Shapley inconsistency values, in: P. Doherty, J. Mylopoulos, C. Welty (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the 10th International Conference (KR06), AAAI Press, 2006, pp. 249–259.

[2] G. Qi, W. Liu, D. Bell, Measuring conflict and agreement between two prioritized belief bases, in: L.P. Kaelbling, A. Saffiotti (Eds.), Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI05), July 30–August 5, 2005, Professional Book Center, Edinburgh, UK, 2005, pp. 552–557.

[3] 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), Hammamet, Tunisia, October 31–November 2, 2007, in: LNCS, vol. 4724, Springer, 2007, pp. 30–41.

[4] K. Mu, Z. Jin, R. Lu, W. Liu, Measuring inconsistency in requirements specifications, in: L. Godo (Ed.), Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU2005), in: LNCS, vol. 3571, Springer-Verlag, 2005, pp. 440–451.
[5] K. Mu, J. Hong, Z. Jin, W. Liu, From inconsistency handling to non-canonical requirements management: A logical perspective, Int. J. Approx. Reason. 54 (1) (2013) 109–131.
[6] K. Mu, Z. Jin, W. Liu, D. Zowghi, B. Wei, Measuring the significance of inconsistency in the viewpoints framework, Sci. Comput. Program. 78 (9) (2013) 1572–1599.
[7] D.P. Muiño, Measuring and repairing inconsistency in probabilistic knowledge bases, Int. J. Approx. Reason. 52 (6) (2011) 828–840.
[8] K. McAreavey, 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.
[9] J. Grant, Classifications for inconsistent theories, Notre Dame J. Form. Log. 19 (3) (1978) 435–444.
[10] L. Bertossi, A. Hunter, T. Schaub, Introduction to inconsistency tolerance, in: L. Bertossi, A. Hunter, T. Schaub (Eds.), Inconsistency Tolerance, in: Lecture Notes in Computer Science, vol. 3300, Springer, 2004, pp. 1–16.
[11] J. Grant, A. Hunter, Measuring inconsistency in knowledge bases, J. Intell. Inf. Syst. 27 (2) (2006) 159–184.
[12] J. Grant, A. Hunter, Analysing inconsistent first-order knowledge bases, Artif. Intell. 172 (8–9) (2008) 1064–1093.
[13] A. Hunter, S. Konieczny, On the measure of conflicts: Shapley inconsistency values, Artif. Intell. 174 (14) (2010) 1007–1026.
[14] G. Xiao, Z. Lin, Y. Ma, G. Qi, Computing inconsistency measurements under multi-valued semantics by partial max-sat solvers, in: F. Lin, U. Sattler, M. Truszczynski (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the 10th International Conference (KR2010), AAAI Press, 2010, pp. 340–349.
[15] A. Hunter, Logical comparison of inconsistent perspectives using scoring functions, Knowl. Inf. Syst. 6 (5) (2004) 528–543.
[16] A. Hunter, S. Konieczny, Measuring inconsistency through minimal inconsistent sets, in: G. Brewka, J. Lang (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference (KR08), AAAI Press, 2008, pp. 358–366.
[17] K. Mu, W. Liu, Z. Jin, A general framework for measuring inconsistency through minimal inconsistent sets, Knowl. Inf. Syst. 27 (1) (2011) 85–114.
[18] 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 (7) (2011) 978–999.
[19] G. Xiao, Y. Ma, Inconsistency measurement based on variables in minimal unsatisfiable subsets, in: L.D. Raedt, C. Bessière, D. Dubois, P. Doherty, P. Frasconi, F. Heintz, P.J.F. Lucas (Eds.), ECAI 2012 – 20th European Conference on Artificial Intelligence, Montpellier, France, August 27–31, 2012, in: Frontiers in Artificial Intelligence and Applications, vol. 242, IOS Press, 2012, pp. 864–869.
[20] K. Knight, Measuring inconsistency, J. Philos. Log. 31 (1) (2002) 77–98.
[21] K. Knight, Two information measures for inconsistent sets, J. Log. Lang. Inf. 12 (2) (2003) 227–248.
[22] S. Konieczny, J. Lang, P. Marquis, Quantifying information and contradiction in propositional logic through epistemic actions, in: G. Gottlob, T. Walsh (Eds.), Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI2003), Morgan Kaufmann, 2003, pp. 106–111.
[23] J. Delgrande, D. Dubois, J. Lang, Iterated revision as prioritized merging, in: P. Doherty, J. Mylopoulos, C.A. Welty (Eds.), Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2–5, 2006, AAAI Press, 2006, pp. 210–220.
[24] G. Qi, W. Liu, D.A. Bell, Merging stratified knowledge bases under constraints, in: Proceedings of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, Boston, MA, USA, July 16–20, 2006, AAAI Press, 2006, pp. 281–286.
[25] K. Mu, W. Liu, Z. Jin, R. Lu, A. Yue, D.A. Bell, Handling inconsistency in distributed software requirements specifications based on prioritized merging, Fundam. Inform. 91 (3–4) (2009) 631–670.
[26] G. Brewka, Preferred subtheories: An extended logical framework for default reasoning, in: N.S. Sridharan (Ed.), Proceedings of the 11th International Joint Conference on Artificial Intelligence, Detroit, MI, USA, August 1989, Morgan Kaufmann, 1989, pp. 1043–1048.
[27] S. Benferhat, D. Dubois, H. Prade, A local approach to reasoning under inconsistency in stratified knowledge bases, in: Lecture Notes in Computer Science, vol. 946, Springer, 1995, pp. 36–43.
[28] G. Priest, Minimally inconsistent LP, Stud. Log. 50 (1) (1991) 321–331.
[29] G. Brewka, A rank based description language for qualitative preferences, in: R.L. de Mántaras, L. Saitta (Eds.), Proceedings of the 16th European Conference on Artificial Intelligence, ECAI'2004, Valencia, Spain, August 22–27, 2004, IOS Press, 2004, pp. 303–307.
[30] A. Hunter, S. Konieczny, Approaches to measuring inconsistent information, in: L. Bertossi, A. Hunter, T. Schaub (Eds.), Inconsistency Tolerance, in: Lecture Notes in Computer Science, vol. 3300, Springer, 2005, pp. 191–236.
[31] M. Thimm, Measuring inconsistency in probabilistic knowledge bases, in: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI'09), Montreal, Canada, June 2009, AUAI Press, 2009, pp. 530–537.
[32] C. Papadimitriou, Computational Complexity, Addison–Wesley, Massachusetts, 1994.
[33] Y. Ma, G. Qi, G. Xiao, P. Hitzler, Z. Lin, Computational complexity and anytime algorithm for inconsistency measurement, Int. J. Softw. Inform. 4 (1) (2010) 3–21.
[34] T. Eiter, G. Gottlob, On the complexity of propositional knowledge base revision, updates, and counterfactuals, Artif. Intell. 57 (2–3) (1992) 227–270.
[35] S. Benferhat, S. Kaci, D.L. Berre, M.-A. Williams, Weakening conflicting information for iterated revision and knowledge integration, in: B. Nebel (Ed.), Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, August 4–10, 2001, Morgan Kaufmann, 2001, pp. 109–118.
[36] S. Benferhat, S. Kaci, D.L. Berre, M.-A. Williams, Weakening conflicting information for iterated revision and knowledge integration, Artif. Intell. 153 (1–2) (2004) 339–371.
[37] K. Mu, W. Liu, Z. Jin, Measuring the blame of each formula for inconsistent prioritized knowledge bases, J. Log. Comput. 22 (3) (2012) 481–516.