Measuring Consistency Gain and Information Loss in Stepwise Inconsistency Resolution

John Grant[1] and Anthony Hunter[2]

[1] Department of Mathematics, Towson University, Towson, MD 21252, USA
and Department of Computer Science, University of Maryland,
College Park, MD 20742, USA
grant@cs.umd.edu

[2] Department of Computer Science, University College London,
Gower Street, London WC1E 6BT, UK
a.hunter@cs.ucl.ac.uk

Abstract. Inconsistency is a usually undesirable feature of many kinds of data and knowledge. But altering the information in order to make it less inconsistent may result in the loss of information. In this paper we analyze this trade-off. We review some existing proposals and make new proposals for measures of inconsistency and information. We prove that in both cases the various measures are all pairwise incompatible. Then we introduce the concept of stepwise inconsistency resolution and show what happens in case an inconsistency resolution step applies a deletion, a weakening, or a splitting operation.

1 Introduction

Inconsistency, and deciding how to deal with it, is a well-recognized problem in many areas of computer science including data and knowledge engineering, software engineering, robotics, and natural language. Often it is not possible to determine with high confidence which items of data or knowledge are incorrect. It might be that to find this out would cost more than the information is actually worth. Or it might be that it is just not possible to acquire this information. In these situations, it may however be useful to delete or update items of information that are involved in inconsistencies based on the nature of those inconsistencies. But since it is often unclear which items of information should be changed, the process of inconsistency resolution can result in a gain in the degree of consistency, but at the price of a loss of information.

In this paper, we propose the use of inconsistency and information measures to take account of this trade-off. We start by investigating what are essential properties of inconsistency and information measures. We propose three requirements in both cases and consider various definitions, mostly ones previously proposed. Each proposal has some rationale, so it is worthwhile to investigate their compatibility with one another. We will show that in a well-defined sense each measure is incompatible with every other measure, and this goes both for inconsistency measures and information measures. These results suggest that there does not exist a single inconsistency measure or information measure that coincides with intuition in general. Nonetheless, the framework for inconsistency and information measures is potentially useful for choosing measures according to specific applications.

To illustrate some of the key issues in stepwise inconsistency resolution, we consider the following example. Let $K = \{a, \neg a \wedge \neg b \wedge \neg c, b, d\}$. K has two minimal inconsistent subsets: $M_1 = \{a, \neg a \wedge \neg b \wedge \neg c\}$ and $M_2 = \{\neg a \wedge \neg b \wedge \neg c, b\}$; and two maximal consistent subsets $N_1 = \{a, b, d\}$ and $N_2 = \{\neg a \wedge \neg b \wedge \neg c, d\}$. As we want to show how to reduce the inconsistency of K in a stepwise fashion, one formula at a time, we will apply three inconsistency resolution functions: delete a formula, weaken a formula, and split a formula.

– **Deletion** We delete a formula that is in a minimal inconsistent subset. Thus we can delete either $\neg a \wedge \neg b \wedge \neg c$ or a or b. In the first case, since $\neg a \wedge \neg b \wedge \neg c$ is in both minimal inconsistent subsets, the result is consistent. This is the most drastic of the three options because this operation loses the most information.
– **Weakening** We change a formula to another formula logically implied by it. Typically, we add a disjunct or change a conjunction to a disjunction. For instance, we can weaken $\neg a \wedge \neg b \wedge \neg c$ to $(\neg a \vee \neg b) \wedge \neg c$ or $\neg a \vee \neg b \vee \neg c$. We can weaken a to $a \vee b$ or even $a \vee \neg a$, and so on. While this operation may reduce the number of minimal inconsistent subsets, the size of the minimal inconsistent subsets may rise, as seen here, where the first weakening results in one minimal inconsistent subset $\{a, (\neg a \vee \neg b) \wedge \neg c, b\}$.
– **Splitting** We split a formula into its conjuncts. This may isolate the really problematic conjuncts. For instance, we can split $\neg a \wedge \neg b \wedge \neg c$ into $\neg a$, $\neg b$, and $\neg c$. In this case, we get a new knowledgebase $\{a, \neg a, b, \neg b, \neg c, d\}$ that is still inconsistent, though by some inconsistency measures it is less inconsistent. Also, this allows us at a later step to delete just the portion of the conjunction involved in the inconsistency.

In an inconsistent knowledgebase, any formula involved in an inconsistency can be selected for one of the resolution operations (of deletion, weakening or splitting). So there is a question of how to choose a formula and which operation to apply. In general, inconsistency and information measures offer possible answers to this question. Our guiding principle is to minimize information loss while reducing inconsistency as we resolve an inconsistent knowledgebase by stepwise resolution.

2 Preliminary Definitions

We assume a propositional language $\mathcal{L}$ of formulae composed from a set of atoms $\mathcal{A}$ and the logical connectives $\wedge$, $\vee$, $\neg$. We use ϕ and ψ for arbitrary formulae and α and β for atoms. All formulae are assumed to be in conjunctive normal form. Hence every formula ϕ has the form $\psi_1 \wedge \ldots \wedge \psi_n$, where each ψ_i, $1 \leq i \leq n$, has the form $\beta_{i1} \vee \ldots \vee \beta_{im}$, where each β_{ik}, $1 \leq k \leq m$ is a literal (an atom or negated atom). A knowledgebase K is a finite set of formulae. We let $\vdash$ denote the classical consequence relation, and write $K \vdash \bot$ to denote that K is inconsistent. Logical equivalence is defined in the usual way: $K \equiv K'$ iff $K \vdash K'$ and $K' \vdash K$. We find it useful to define also a stronger notion of equivalence we call b(ijection)-equivalence as follows.

Knowledgebase K is b(ijection)-equivalent to knowledgebase K', denoted $K \equiv_b K'$ iff there is a bijection $f : K \rightarrow K'$ such that for all $\phi \in K$, ϕ is logically equivalent to $f(\phi)$. For example, $\{a, b\}$ is logically equivalent but not b(ijection)-equivalent to $\{a \wedge b\}$. We write $\mathcal{R}^{\geq 0}$ for the set of nonnegative real numbers and $\mathcal{K}$ for the set of all knowledgebases (in some presumed language $\mathcal{L}$).

For a knowledgebase K, $\mathsf{MI}(K)$ is the set of minimal inconsistent subsets of K, and $\mathsf{MC}(K)$ is the set of maximal consistent subsets of K. Also, if $\mathsf{MI}(K) = \{M_1, ..., M_n\}$ then $\mathsf{Problematic}(K) = M_1 \cup ... \cup M_n$, and $\mathsf{Free}(K) = K \setminus \mathsf{Problematic}(K)$. So $\mathsf{Free}(K)$ contains the formulae in K that are not involved in any inconsistency and $\mathsf{Problematic}(K)$ contains the formulae in K that are involved in at least one inconsistency. The set of formulae in K that are individually inconsistent is given by the function $\mathsf{Selfcontradictions}(K) = \{\phi \in K \mid \{\phi\} \vdash \bot\}$). In the next section we will use these functions in definitions for syntactic measures of inconsistency.

α	T	T	T	B	B	B	F	F	F
β	T	B	F	T	B	F	T	B	F
$\alpha \vee \beta$	T	T	T	T	B	B	T	B	F
$\alpha \wedge \beta$	T	B	F	B	B	F	F	F	F
$\neg\alpha$	F	F	F	B	B	B	T	T	T

Fig. 1. Truth table for three valued logic (3VL). This semantics extends the classical semantics with a third truth value, B, denoting "contradictory". Columns 1, 3, 7, and 9, give the classical semantics, and the other columns give the extended semantics.

The corresponding semantics uses Priest's three valued logic (3VL) [11] with the classical two valued semantics augmented by a third truth value denoting inconsistency. The truth values for the connectives are defined in Figure 1. An interpretation i is a function that assigns to each atom that appears in K one of three truth values: i : $\mathsf{Atoms}(K) \rightarrow \{F, B, T\}$. For an interpretation i it is convenient to separate the atoms into two groups, namely the ones that are assigned a classical truth value and the ones that are assigned B.

$$\mathsf{Binarybase}(i) = \{\alpha \mid i(\alpha) = T \text{ or } i(\alpha) = F\}$$

$$\mathsf{Conflictbase}(i) = \{\alpha \mid i(\alpha) = B\}$$

For a knowledgebase K we define the models as the set of interpretations where no formula in K is assigned the truth value F: $\mathsf{Models}(K) = \{i \mid \text{ for all } \phi \in K, i(\phi) = T \text{ or } i(\phi) = B\}$ Then, as a measure of inconsistency for K we define

$$\mathsf{Contension}(K) = \mathsf{Min}\{|\mathsf{Conflictbase}(i)| \mid i \in \mathsf{Models}(K)\}$$

So the contension gives the minimal number of atoms that need to be assigned B in order to get a 3VL model of K.

Example 1. For $K = \{a, \neg a, a \vee b, \neg b\}$, there are two models of K, i_1 and i_2, where $i_1(a) = B$, $i_1(b) = B$, $i_2(a) = B$, and $i_2(b) = F$. Therefore, $\mathsf{Conflictbase}(i_1) = 2$ and $\mathsf{Conflictbase}(i_2) = 1$. Hence, $\mathsf{Contension}(K) = 1$.

Finally, we consider some useful definitions based on the notion of implicants. A consistent set of literals X is an **implicant** for a knowledgebase K iff for each $\phi \in K$, $X \vdash \phi$. A minimal implicant is called a **prime implicant**. For example, for $K = \{a, \neg b \vee c\}$, the prime implicants are $X_1 = \{a, \neg b\}$ and $X_2 = \{a, c\}$. A **proxy** for K is a set of literals X such that X is a prime implicant of a maximal consistent subset of K. Let the set of proxies for K (denoted $\mathsf{Proxies}(K)$) be defined as follows.

$$\mathsf{Proxies}(K) = \{X \mid X \text{ is a prime implicant of } K' \in \mathsf{MC}(K)\}$$

For example, for $K = \{a, \neg a, b \vee c\}$, $\mathsf{Proxies}(K) = \{\{a, b\}, \{\neg a, b\}, \{a, c\}, \{\neg a, c\}\}$.

We see that each proxy represents an "interpretation" of the possible literals that hold, and so the number of proxies rises by increasing the number of disjuncts in any formula, and by increasing the number of conflicting formulae. The cardinality of each proxy rises with the amount of information in each alternative, and so adding conjuncts to a formula will increase the size of one or more proxies (as long as the conjunction is consistent).

3 Inconsistency and Information Measures

In this section, we study inconsistency and information measures. We consider both existing and new proposals. Our main result is that for both inconsistency measures and information measures, the various measures are incompatible with one another. This result strongly implies that unlike some other intuitive concepts, such as the concept of effective computability, where different definitions using recursion, λ-calculus, and Turing machines are equivalent, both inconsistency measure and information measure are too elusive to be captured by a single definition. Additionally, for information measures we also consider various plausible constraints and investigate which measures satisfy them.

3.1 Inconsistency Measures for Knowledgebases

An inconsistency measure assigns a nonnegative real value to every knowledgebase. We make three requirements for inconsistency measures. The constraints ensure that all and only consistent knowledgebases get measure 0, the measure is monotonic for subsets, and the removal of a formula that does not participate in an inconsistency leaves the measure unchanged.

Definition 1. *An inconsistency measure $I : \mathcal{K} \rightarrow \mathcal{R}^{\geq 0}$ is a function such that the following three conditions hold:*

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

3. For all $\alpha \in \mathsf{Free}(K), (I(K) = I(K \backslash \{\alpha\})$.

The above requirements are taken from [3] where (1) is called *consistency*, (2) is called *monotony*, and (3) is called *free formula independence*.

Next we introduce five inconsistency measures: the rationale for each is given below.

Definition 2. *For a knowledgebase* K*, the inconsistency measures* I_C*,* I_P*,* I_B*,* I_S*, and* I_R *are s.t.*

- $I_C(K) = |\mathsf{MI}(K)|$
- $I_M(K) = (|\mathsf{MC}(K)| + |\mathsf{Selfcontradictions}(K)|) - 1$
- $I_P(K) = |\mathsf{Problematic}(K)|$
- $I_B(K) = \mathsf{Contension}(K)$
- $I_Q(K) = \begin{cases} 0 & \text{if } K \text{ is consistent} \\ \sum_{X \in \mathsf{MI}(K)} \frac{1}{|X|} & \text{otherwise} \end{cases}$

We explain the measures as follows: $I_C(K)$ counts the number of minimal inconsistent subsets of K; $I_M(K)$ counts the sum of the number of maximal consistent subsets together with the number of contradictory formulae but 1 must be subtracted to make $I(K) = 0$ when K is consistent; $I_P(K)$ counts the number of formulae in minimal inconsistent subsets of K; $I_B(K)$ counts the minimum number of atoms that need to be assigned B amongst the 3VL models of K; and I_Q computes the weighted sum of the minimal inconsistent subsets of K, where the weight is the inverse of the size of the minimal inconsistent subset (and hence smaller minimal inconsistent subsets are regarded as more inconsistent than larger ones). Each of these measures satisfies the definition of being an inconsistency measure (i.e. Definition 1).

There is a rationale for each inconsistency measure. We cannot require these differently defined measures to give identical numerical values but it would be reasonable to assume that at least some of them place the knowledgebases in the same order with respect to inconsistency. Define I_x and I_y to be *order-compatible* if for all knowledgebases K_1 and K_2, $I_x(K_1) < I_x(K_2)$ iff $I_y(K_1) < I_y(K_2)$ and *order-incompatible* otherwise. The next theorem shows that order-compatibility doesn't hold for any pair of the inconsistency measures we have defined, leading us to think that inconsistency is too elusive a concept to be captured in a single measure.

Theorem 1. [3] I_C*,* I_M*,* I_P*,* I_B*, and* I_Q *are pairwise order-incompatible.*

Although the five inconsistency measures are quite different, four of them give identical results on bijection-equivalent knowledge bases.

Proposition 1. *If* $K \equiv_b K'$ *then* $I_Z(K) = I_Z(K')$ *for* $Z \in \{C, M, P, Q\}$.

Interestingly, b-equivalence does not guarantee equality for I_B. The problem is with self-contradictions. For instance, if $K = \{a \wedge \neg a\}$ and $K' = \{a \wedge \neg a \wedge b \wedge \neg b\}$, then $K \equiv_b K'$, but $I_B(K) = 1 \neq I_B(K') = 2$.

[3] All proofs and additional references are given in a technical report available at www.cs.ucl.ac.uk/staff/a.hunter/papers/stepwise.pdf.

The use of minimal inconsistent subsets, such as I_C, I_P, and I_Q, and the use of maximal consistent subsets such as I_M, have been proposed previously for measures of inconsistency [2, 4]. The idea of a measure that is sensitive to the number of formulae to produce an inconsistency eminates from Knight [8] in which the more formulae needed to produce the inconsistency, the less inconsistent the set. As explored in [4], this sensitivity is obtained with I_Q. Another approach involves looking at the proportion of the language that is touched by the inconsistency such as I_B. Whilst model-based techniques have been proposed before for measures of inconsistency, I_B is a novel proposal since it is based on three-valued logic, and as such, is simpler than the ones based on four-valued logic (e.g. [5]).

3.2 Information Measures for Knowledgebases

Another dimension to analysing inconsistency is to ascertain the amount of information in a knowledgebase. The following novel proposal for an information measure assigns a nonnegative real number to every knowledgebase. The constraints ensure that the empty set has measure 0, the measure is subset monotonic for consistent knowledgebases, and a consistent knowledgebase that does not contain only tautologies has nonzero measure.

Definition 3. *An* **information measure** $J : \mathcal{K} \rightarrow \mathcal{R}^{\geq 0}$ *is a function such that the following three conditions hold:*

1. *If* $K = \emptyset$ *then* $J(K) = 0$.
2. *If* $K' \subseteq K$, *and* K *is consistent, then* $J(K') \leq J(K)$.
3. *If* K *is consistent and* $\exists \phi \in K$ *such that* ϕ *is not a tautology, then* $J(K) > 0$.

The above definition is a general definition that allows for a range of possible measures to be defined. Next we introduce seven information measures; the rationale for each is given below. We note here that in the definition of J_B we will use the concept of Models as previously defined for 3VL. However, in the case of J_L we will need a model concept using classical 2-valued interpretations. We write $\mathsf{2VModels}(K) = \{i|$ is a 2-valued interpretation and for all $\phi \in K, i(\phi) = T\}$.

Definition 4. *For a knowledgebase K, the information measures* J_A, J_S, J_F, J_C, J_B, J_P, *and* J_L *are such that*

- $J_A(K) = |\mathsf{Atoms}(K)|$
- $J_S(K) = |K|$
- $J_F(K) = |\mathsf{Free}(K)|$
- $J_C(K) = \mathsf{Max}\{\ |M| \mid M \in \mathsf{MC}(K)\}$
- $J_B(K) = \mathsf{Max}\{\ |\mathsf{Binarybase}(i)| \mid i \in \mathsf{Models}(K)\}$
- $J_P(K) = \mathsf{Max}\{\ |X| \mid X \in \mathsf{Proxies}(K)\}$
- $J_L(K) = \log_2 \frac{2^n}{|\bigcup\{\mathsf{2VModels}(K')|K' \in \mathsf{MC}(K)\}|}$ *where* $n = |\mathsf{Atoms}(K)|$ *if* $n \geq 1$, *else* $J_L(K) = 0$.

The first two measures do not actually deal with inconsistency at all: J_A counts the number of atoms and J_S counts the number of formulae. For the other four measures:

J_F counts the number of free formulae; J_C finds the size of the largest maximal consistent subset; J_B finds the maximum number of atoms that need not be assigned B in the 3VL models; J_P finds the size of the largest proxy; and J_L uses an information-theoretic approach that is discussed further at the end of this section. All seven measures are information measures according to Definition 3.

In analogy to inconsistency measures, we can define order-compatibility and order-incompatibility for information measures. Similarly, we find that order-compatibility does not hold for any pair of information measures, leading us to think that information is also too elusive a concept to be captured in a single measure.

Theorem 2. *J_A, J_S, J_F, J_C, J_B, J_P, and J_L are pairwise order-incompatible.*

Next we prove some results concerning information measures followed by some that relate information measures with inconsistency measures.

Proposition 2. *If K is consistent, then $J_S(K) = J_F(K) = J_C(K)$.*

Proposition 3. *If K is a set of literals, then $J_A(K) = J_C(K) = J_P(K)$.*

Proposition 4. *For any knowledgebase K, $J_S(K) - J_F(K) = I_P(K)$.*

Proposition 5. *For any knowledgebase K, $J_A(K) - J_B(K) = I_B(K)$.*

Proposition 6. *No information measure is also an inconsistency measure.*

Since our definition of information measure (i.e. Definition 3) is rather weak we consider additional constraints that can be useful for comparing information measures. For an information measure J, and for any knowledgebases $K, K' \subseteq \mathcal{L}$, we call J:

- (Monotonic) If $K \subseteq K'$, then $J(K) \leq J(K')$.
- (Clarity) For all $\phi \in K$, $J(K) \geq J(K \cup \{\psi\})$, where ψ is the cnf of $\neg\phi$.
- (Equivalence) If K is consistent and $K \equiv K'$, then $J(K) = J(K')$.
- (Bijection-Equivalence) If $K \equiv_b K'$, then $J(K) = J(K')$.
- (Closed) If K is consistent, and $K \vdash \phi$, then $J(K) = J(K \cup \{\phi\})$.
- (Cumulative) If $K \cup \{\phi\}$ is consistent, and $K \not\vdash \phi$, then $J(K) < J(K \cup \{\phi\})$.

A monotonic measure is monotonic even for inconsistent knowledgebases. A clarity measure does not increase when the negation of a formula in the knowledgebase is added. An equivalence measure assigns the same value to logically equivalent consistent knowledgebases. A bijection-equivalence measure (which was first proposed in [8]) has the same value for a pair of knowledgebases when the formulae are pairwise equivalent. A closed measure (which was first proposed in [9]) does not have increased information for a consistent knowledgebase when entailed formulae are added. A cumulative measure (which was first proposed in [9]) has increased information for a consistent knowledgebase when a non-entailed formula is added that is consistent with it. We note that if an information measure has the equivalence property then it is closed because if $K \vdash \phi$ then $K \equiv K \cup \{\phi\}$.

Theorem 3. *Figure 2 indicates the constraints that hold for each of the information measures J_A, J_S, J_F, J_C, J_B, J_P, and J_L.*

	J_A	J_S	J_F	J_C	J_B	J_P	J_L
Monotonic	×	×		×		×	
Clarity	×		×		×	×	×
Equivalence						×	×
B-Equivalence		×	×	×		×	×
Closed						×	×
Cumulative		×	×	×		×	×

Fig. 2. Summary of constraints that hold (indicated by ×) for particular information measures

Depending on which constraints one considers important, one may choose from those measures that satisfy them. In particular, J_P satisfies all seven constraints.

The J_A, J_S, J_F, and J_C measures are simple syntactic measures that have been considered in some form before (see for example [2] for a discussion)). However, the J_B and J_P are novel proposals for information measures. There have also been proposals for measures of information for propositional logic based on Shannon's information theory (see for example [6]). Essentially, these measures consider the number of models of the set of formulae (the less models, the more informative the set), and in case the set of formulae is consistent, the result is intuitive. However, when the set is inconsistent, the set is regarded as having null information content. To address the need to consider inconsistent information, Lozinskii proposed a generalization of the information-theoretic approach to measuring information [9] that we called J_L earlier.

4 Stepwise Inconsistency Resolution

Generally, when a knowledgebase is inconsistent, we would like to reduce its inconsistency value, preferably to 0. The problem is that a reduction in inconsistency may lead to a corresponding reduction in information. Consider, for instance, J_S. This measure counts the number of formulae in the knowledgebase. Hence any deletion reduces it. Our goal is to reduce inconsistency with as little information loss as possible, a task that depends on the choice of both the inconsistency measure and the information measure.

We start by formally defining the three functions that we allow in the process of inconsistency resolution. They appear to be representative of all options. These operations will be applied to inconsistent knowledgebases.

Definition 5. *An inconsistency resolution function* irf*, is one of the following three functions* $d(\phi)$ *or* $w(\phi, \psi)$ *or* $s(\phi)$ *where* $\phi \in K$*:*

- *(Deletion)* $d(\phi) = K \setminus \{\phi\}$.
- *(Weakening)* $w(\phi, \psi) = (K \setminus \{\phi\}) \cup \{\psi\}$ *where* $\phi \vdash \psi$.
- *(Splitting)* $s(\phi) = (K \setminus \{\phi\}) \cup \{\phi_1, \ldots, \phi_n\}$ *where* $\phi_1, \ldots, \phi_n$ *are the conjuncts in* ϕ.

Then $\mathsf{irf}(K)$ *is the knowledgebase obtained by applying* irf *to* K. *Also* $\mathsf{irf}(K) = K$ *in case* $\phi \notin K$.

In the stepwise inconsistency resolution process we will usually have multiple applications of such functions. A stepwise resolution function sequence (abbr. *function sequence*) $\mathcal{F} = \langle \mathsf{irf}_1, \ldots, \mathsf{irf}_n \rangle$ is a sequence of such functions. A stepwise inconsistency resolution knowledgebase sequence (abbr. *knowledgebase sequence*) $\mathcal{K}_{\mathcal{F}} = \langle K_0, \ldots, K_n \rangle$ is a sequence of knowledgebases obtained by using $\mathcal{F}$ such that K_0 is the initial knowledgebase and $\mathsf{irf}_i(K_{i-1}) = K_i$ for $1 \leq i \leq n$. We also write $\mathcal{F}(K_0) = K_n$ and observe that $K_n = \mathsf{irf}_n(\ldots \mathsf{irf}_1(K_0) \ldots)$.

The goal of stepwise inconsistency resolution is to reduce the inconsistency of the knowledgebase. Next we define a simple way to measure the reduction . We will be interested in applying this definition to the case where $\mathcal{F}(K) = K'$ for some function sequence $\mathcal{F}$.

Definition 6. *Given an inconsistency measure I, an inconsistency resolution measure $R_I : \mathcal{K} \times \mathcal{K} \to \mathcal{R}$ is defined as follows:*

$$R_I(K, K') = I(K) - I(K')$$

For illustration we give two examples. The example given in Figure 3 corresponds to deletion, and Example 2 corresponds to splitting a formula.

	α					
	a	$\neg a \wedge b$	$\neg b \vee c$	$\neg c$	$c \vee d$	$\neg d$
$R_{I_C}(K, K \setminus \{\alpha\})$	1	2	1	2	1	1
$R_{I_M}(K, K \setminus \{\alpha\})$	1	3	0	4	3	3
$R_{I_P}(K, K \setminus \{\alpha\})$	1	3	1	4	2	2
$R_{I_B}(K, K \setminus \{\alpha\})$	1	1	0	1	0	0
$R_{I_Q}(K, K \setminus \{\alpha\})$	3/6	5/6	2/6	4/6	2/6	2/6

Fig. 3. Illustration of resolution measures applied to knowledgebases obtained by deleting a formula from the knowledgebase $K = \{a, \neg a \wedge b, \neg b \vee c, \neg c, c \vee d, \neg d\}$. Here we see that according to I_P, $\neg c$ is the optimal choice for deletion, while for I_Q, it is $\neg a \wedge b$.

Example 2. Let $K = \{a, \neg a \wedge \neg b, b\}$. Splitting K by applying $s(\neg a \wedge \neg b)$ we obtain $K' = \{a, \neg a, b, \neg b\}$. Here we see that splitting does not reduce inconsistency according to any of the five inconsistency measures. Indeed, for several measures it causes an increase in inconsistency .

$$\begin{array}{l} R_{I_C}(K, (K \setminus \{\neg a \wedge \neg b\}) \cup \{\neg a, \neg b\}) = 0 \\ R_{I_M}(K, (K \setminus \{\neg a \wedge \neg b\}) \cup \{\neg a, \neg b\}) = -2 \\ R_{I_P}(K, (K \setminus \{\neg a \wedge \neg b\}) \cup \{\neg a, \neg b\}) = -1 \\ R_{I_B}(K, (K \setminus \{\neg a \wedge \neg b\}) \cup \{\neg a, \neg b\}) = 0 \\ R_{I_Q}(K, (K \setminus \{\neg a \wedge \neg b\}) \cup \{\neg a, \neg b\}) = 0 \end{array}$$

Some simple observations concerning the R_I measure are the following: (1) If $\phi \notin K$, then $R_I(K, K \setminus \{\phi\}) = 0$ and (2) If $\phi \in \mathsf{Free}(K)$ then $R_I(K, K \setminus \{\phi\}) = 0$.

In the stepwise resolution process we try to minimize the loss of information as well. For this reason we now define a way to measure the loss of information.

Definition 7. *Given an information measure J, an information loss measure $R_J : \mathcal{K} \times \mathcal{K} \to \mathcal{R}$ is defined as follows.*

$$R_J(K, K') = J(K) - J(K')$$

Our general goal is to simultaneously maximize R_I and minimize R_J. In the following subsections we consider some of the issues for each of the options we have (i.e. for deletion, for weakening, and for splitting).

4.1 Inconsistency Resolution by Deletion

Deletion is the simplest, and yet most drastic, of the options we have for dealing with inconsistency. In terms of deciding of how to proceed, if deletion is the only function used, it is just a matter of choosing a formula to delete at each step. The following result describes the possibilities for both R_I and R_J when K' is obtained from K by a single deletion.

Theorem 4. *Let K' be obtained from an inconsistent K by deleting a single formula.*
(a) For all 5 inconsistency measures $R_I(K, K') \geq 0$.
(b) For the information measures J_F, J_B and J_L, $R_J(K, K')$ may be negative; in the other cases $R_J(K, K')$ is a nonnegative integer.

The following result follows immediately from the second constraint of an information measure and will be useful in narrowing the knowledgebases that need to be considered for minimal information loss when inconsistency resolution is done by deletions.

Proposition 7. *If K is consistent then $R_J(K, K \setminus \{\phi\}) \geq 0$.*

This result shows that once we delete enough formulae from an inconsistent knowledgebase to make it consistent (and thereby make any inconsistency measure 0), we might as well stop because additional deletions may only cause information loss. This gives the following result.

Corollary 1. *Suppose that stepwise inconsistency resolution is done by deletions only. To find a consistent knowledgebase with minimal information loss (i. e. where $R_J(K, K')$ is minimal) it suffices to consider only those function sequences $\mathcal{F}$ where $\mathcal{F}(K) \in \mathsf{MC}(K)$.*

4.2 Inconsistency Resolution by Weakening

In this subsection we investigate the case where the inconsistency of a knowledgebase is resolved by using weakenings only. Thus we start with an inconsistent knowledgebase K and by applying one or more weakenings we obtain a consistent K'. Our concern here is what happens to the information measure during this process. In order to analyze this situation we will exclude the case where a formula is weakened by using an atom not in K such as by applying a disjunction with such an atom. We do this because it does not seem reasonable to change the language of the knowledgebase when our purpose is to weaken it for consistency. Also, by excluding this case we make sure that the information measure cannot become arbitrarily large by simply taking bigger and bigger disjuncts with new atoms.

Our result is summarized in the following theorem.

Theorem 5. *Let K be an inconsistent knowledgebase that is transformed to a consistent knowledgebase K' by one or more weakenings without introducing any atom not already in K. Then*

1. $J_A(K') \leq J_A(K)$.
2. $J_S(K') \leq J_S(K)$.
3. $J_F(K') \geq J_F(K)$.
4. $J_C(K') \geq J_C(K)$.
5. *No inequality holds between $J_B(K')$ and $J_B(K)$.*
6. $J_P(K') \leq J_P(K)$.
7. $J_L(K') \geq J_L(K)$.

4.3 Inconsistency Resolution using Splitting

Here we consider what happens when splitting is applied. First we note that unlike deletion and weakening, splitting by itself cannot resolve inconsistencies. Hence splitting must be used in conjunction with deletion or weakening. We start by considering what happens when just splitting is applied. Just as in the case of deletions and weakenings, we split only formulae in $\mathsf{Problematic}(K)$.

Theorem 6. *Let K' be obtained from an inconsistent knowledgebase K by splitting a single formula in $\mathsf{Problematic}(K)$. Then*

(a)
1. $I_C(K') \geq I_C(K)$,
2. $I_M(K') \geq I_M(K)$,
3. $I_P(K') \geq I_P(K)$,
4. $I_B(K') = I_B(K)$,
5. *No inequality holds between $I_Q(K')$ and $I_Q(K)$.*

(b)
1. $J_A(K') = J_A(K)$,
2. $J_S(K') > J_S(K)$,
3. $J_F(K') \geq J_F(K)$,
4. $J_C(K') \geq J_C(K)$,
5. $J_B(K') = J_B(K)$,
6. $J_P(K') = J_P(K)$
7. *No inequality holds between $J_L(K')$ and $J_L(K)$.*

This theorem shows that splitting decreases neither inconsistency nor information (except possibly for I_Q and J_L), and for some measures it increases both. Anyway, as pointed out earlier, splitting must be combined with another operation to eliminate inconsistency.

5 Discussion

In general, inconsistency resolution should be guided by the aim of decreasing inconsistency without excessive loss of information. However, there is a trade-off between the amount to which inconsistency is decreased and the amount of information loss that can be accepted. Futhermore, there can be numerous choices over what resolution steps to take at any state of the knowledgebase.

A common criterion is that some or all operations are not permitted on some formulae. Alternatively, there may be a preference ordering over the formulae such that the less preferred formulae should be considered for being subject to a resolution operation before the more preferred formulae. However, in situations, where two or more formulae can be subjected to a resolution operation, the use of inconsistency and information measures may help in making a choice.

Turning to the question of which measures to use, this depends on the application and the users involved. If they all agree to use specific measures in advance, then that could be their prerogative. However, in general, when agents discuss specific options for stepwise resolution, they may also need to discuss on a stepwise basis which measures to take into account and why.

In this paper, we have clarified the space of inconsistency and information measures and then shown how a wide variety of proposals conform to these general definitions. It is surprising that all different measures are incompatible with one another. We have also shown how inconsistency and information measures can be used to direct stepwise resolution of inconsistency so that inconsistency can be decreased whilst minimising information loss.

References

1. J Grant. Classifications for inconsistent theories. *Notre Dame Journal of Formal Logic*, 19:435–444, 1978.
2. A Hunter and S Konieczny. Approaches to measuring inconsistent information. In *Inconsistency Tolerance*, volume 3300 of *LNCS*, pages 189–234. Springer, 2004.
3. A Hunter and S Konieczny. Shapley inconsistency values. In *Proceedings of KR'06*, pages 249–259. AAAI Press, 2006.
4. A Hunter and S Konieczny. Measuring inconsistency through minimal inconsistent sets. In *Proceedings of KR'08*, pages 358–366. AAAI Press, 2008.
5. A Hunter. Measuring inconsistency in knowledge via quasi-classical models. In *Proc of AAAI'02*, pages 68–73. MIT Press, 2002.
6. J. Kemeny. A logical measure function. *J of Symbolic Logic*, 18:289–308, 1953.
7. S Konieczny, J Lang, and P Marquis. Quantifying information and contradiction in propositional logic through epistemic actions. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 106–111, 2003.
8. K Knight. Measuring inconsistency. *J of Philosophical Logic*, 31:77–98, 2001.
9. E. Lozinskii. Resolving contradictions: A plausible semantics for inconsistent systems. *Journal of Automated Reasoning*, 12:1–31, 1994.
10. K Mu, Z Jin, R Lu, and W Liu. Measuring inconsistency in requirements engineering. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'05)*, volume 3571 of *LNAI*, pages 440–451. Springer, 2005.
11. G Priest. Logic of paradox. *Journal of Philosophical Logic*, 8:219–241, 1979.