Restricted Access Logics for Inconsistent Information

Dov Gabbay and Anthony Hunter

Department of Computing, Imperial College
180, Queen's Gate, London, SW7 2BZ, UK

Abstract. For practical reasoning with classically inconsistent information, desiderata for an appropriate logic L could include (1) it is an extension of classical logic - in the sense that all classical tautologies are theorems of L, and (2) contradictions do not trivialize L - in the sense that ex falso quodlibet does not hold. Two ways of realizing the second desideratum, for any database that may be inconsistent, include (A) take weaker than classical proof rules, but use all the data, or (B) take all the classical proof rules, but restrict the access of the data to the proof rules. The problem with adopting option (A) is that desideratum (1) is then not realizable. In this paper, we pursue option (B) by adding extra conditions on the proof rules to stop certain subsets of the data using the classical proof rules. To facilitate the presentation, we use the approach of Labelled Deductive Systems - formulae are labelled, and proof rules defined to manipulate both the formulae and the labels. The extra conditions on the proof rules are then defined in terms of the labels. This gives us a class of logics, called restricted access logics, that meet the desiderata above.

1 Introduction

For practical reasoning, it is often difficult and inappropriate to maintain a consistent database. Unfortunately, this creates problems if we are to use a logic with such a database. For classical logic, the rule of reductio ad absurdum means that any conclusion can be drawn from the database. This renders the database useless and therefore classical logic is obviously unsatisfactory for this application. A possible solution is to weaken classical logic by dropping reductio ad absurdum. This gives a class of logics called paraconsistent logics such as C_ω (da Costa 1974). However, the weakening of the proof rules means that the connectives in the language do not behave in a classical fashion. For example (taken from Besnard 1991), disjunctive syllogism does not hold, $((\alpha \vee \beta) \wedge \neg\beta) \rightarrow \alpha$ whereas modus ponens does hold. So, for example, α does not follow from Database 1, whereas α does follow from Database 2.

$$\text{Database 1 is } \{(\alpha \vee \beta), \neg\beta\}$$

$$\text{Database 2 is } \{(\neg\beta \rightarrow \alpha), \neg\beta\}$$

There are many similar examples that could be confusing and counter-intuitive for users of such a practical reasoning system.

An alternative, which we explore in this paper, is to not weaken the classical proof rules, but rather to restrict the access of the data to the classical proof rules. The naive version of this is to only allow consistent subsets of the database to be used with the classical proof rules. This would allow intuitive manipulation of the data, but disallow the undesired application of Ex Falso Quodlibet (EFQ) on the inconsistent data.

In the following we define a new consequence relation that captures this reasoning, explore properties of this consequence relation, identify some of its drawbacks and finally, consider interesting variants. The overall objective is to identify logics that allow for the derivation of as many of the classical but non-trivial inferences as possible from inconsistent data. Obviously such inferences are weakly justified but we leave the selection of preferred subsets of these inferences to future work.

2 A logic of restricted access

To present restricted access logics, we use the approach of Labelled Deductive Systems (Gabbay 1991). We assume a set of logical symbols $\{\neg, \rightarrow, \vee, \wedge, \leftrightarrow\}$, a set of atoms $\{\alpha, \beta, \gamma, \cdots\}$, and form the set of formula $\{\alpha \vee \beta, \neg\alpha, \alpha \rightarrow \beta, \cdots\}$ in the usual way. We also assume the natural numbers $\mathbb{N}$, so that if $\mathrm{i} \subseteq \mathbb{N}$, and α is a formula, then i is a label, and $\mathrm{i}{:}\alpha$ is a labelled formula. A database is a set of labelled formulae where each item is labelled with a unique singleton set. For the proof theory, we take the usual natural deduction classical proof rules and amend them to allow handling of labelled formulae. To support this, we define the function h as follows, where $\vdash$ denotes the classical consequence relation,

$$\mathrm{h}(\mathrm{i}) = 1 \text{ iff } \{\alpha \mid \mathrm{j} \subseteq \mathrm{i} \text{ and } \mathrm{j}{:}\alpha \in \Delta\} \not\vdash \bot$$

Essentially $\mathrm{h}(\mathrm{i}) = 1$ if and only if the set of formulae from Δ that are used in the proof, as recorded by the label i, are consistent. Hence the h function is defined with respect to a database Δ. For our first restricted access logic, RAc, the proof rules, are defined as follows, where for $\rightarrow$I, RAA, and $\neg$I, each new assumption is labelled with the empty set. All the rules carry the labels of all the assumptions used in deriving each inference.

$$\wedge\mathrm{I}\ \frac{\mathrm{i}{:}\alpha, \mathrm{j}{:}\beta,\ \mathrm{h}(\mathrm{i}\cup\mathrm{j}) = 1}{\mathrm{i}\cup\mathrm{j}{:}\alpha \wedge \beta} \qquad \wedge\mathrm{E1}\ \frac{\mathrm{i}{:}\alpha \wedge \beta,\ \mathrm{h}(\mathrm{i}) = 1}{\mathrm{i}{:}\alpha} \qquad \wedge\mathrm{E2}\ \frac{\mathrm{i}{:}\alpha \wedge \beta,\ \mathrm{h}(\mathrm{i}) = 1}{\mathrm{i}{:}\beta}$$

$$\rightarrow\mathrm{I}\ \frac{\begin{array}{c}\emptyset{:}\alpha \\ \vdots \\ \mathrm{i}{:}\beta,\ \mathrm{h}(\mathrm{i}) = 1\end{array}}{\mathrm{i}{:}\alpha \rightarrow \beta} \qquad \rightarrow\mathrm{E}\ \frac{\mathrm{i}{:}\alpha,\ \mathrm{j}{:}\alpha \rightarrow \beta,\ \mathrm{h}(\mathrm{i}\cup\mathrm{j}) = 1}{\mathrm{i}\cup\mathrm{j}{:}\beta}$$

$$\mathrm{EFQ}\ \frac{\mathrm{i}{:}\bot,\ \mathrm{h}(\mathrm{i}) = 1}{\mathrm{i}{:}\alpha} \qquad \mathrm{RAA}\ \frac{\begin{array}{c}\emptyset{:}\neg\alpha \\ \vdots \\ \mathrm{i}{:}\bot,\ \mathrm{h}(\mathrm{i}) = 1\end{array}}{\mathrm{i}{:}\alpha}$$

$$\neg\mathrm{I}\ \frac{\begin{array}{c}\emptyset{:}\alpha\\ \vdots\\ \mathrm{i}{:}\bot,\ \mathrm{h(i)} = 1\end{array}}{\mathrm{i}{:}\neg\alpha}
\qquad\qquad
\neg\mathrm{E}\ \frac{\mathrm{i}{:}\alpha,\ \mathrm{j}{:}\neg\alpha,\ \mathrm{h(i\cup j)} = 1}{\mathrm{i\cup j}{:}\bot}$$

In this way, the rules of EFQ, ¬I, and RAA are allowed to use a formula i:⊥ only if the inconsistency was not caused by inconsistent data from the database. In other words, inconsistency must result from the new assumptions labelled with the empty set in concert with the data from the database, or just from the new assumptions labelled with the empty set. We define the consequence relation $\vdash_c$ for RAc logic as follows, where we assume all items in Δ are uniquely labelled with singleton sets.

$$\begin{array}{l}\Delta \vdash_c \alpha \text{ iff } \exists \text{ i such that there is a proof of i:}\alpha\\ \qquad\qquad \text{from } \Delta \text{ using the RAc proof rules}\end{array}$$

Below, we give an example of an acceptable proof of $\neg\beta$ from the database $\Delta = \{\{1\} : \neg\alpha, \{2\} : \neg\alpha \rightarrow \neg\beta, \{3\} : \alpha\}$.

$$\frac{\{1\}{:}\neg\alpha, \{2\} : \neg\alpha \rightarrow \neg\beta, \mathrm{h}(\{1,2\}\ ) = 1}{\{1,2\} : \neg\beta}$$

We now give an example of an unacceptable proof of $\neg\beta$ from the database Δ. In this example, the condition h({1,3}) = 1 does not hold.

$$\frac{\dfrac{\{1\}{:}\neg\alpha, \{3\}{:}\alpha\ \ \mathrm{h}(\{1,3\}) \neq 1}{\{1,3\}{:}\bot,\ \mathrm{h}(\{1,3\}) \neq 1}}{\{1,3\}{:}\neg\beta}$$

As an example of proving a classical tautology using the RAc proof rules, consider $\neg\neg\alpha \rightarrow \alpha$,

$$\frac{\dfrac{\dfrac{\emptyset{:}\neg\neg\alpha\quad \emptyset{:}\neg\alpha\quad \mathrm{h}(\emptyset) = 1}{\emptyset{:}\bot\quad \mathrm{h}(\emptyset) = 1}}{\emptyset{:}\alpha\quad \mathrm{h}(\emptyset) = 1}}{\emptyset{:}\neg\neg\alpha \rightarrow \alpha}$$

Since h(∅) = 1, for any database Δ, it is straightforward to show the tautology with label ∅. To show this, first assume $\emptyset : \neg\neg\alpha$, and show $\emptyset : \alpha$. To do this, assume $\emptyset : \neg\alpha$. So these two assumptions give $\emptyset : \bot$, and hence $\emptyset : \alpha$. Effectively, we have a classical proof where all items are labelled with ∅.

These examples illustrate how the undesirable reasoning with EFQ is limited. We describe a consequence relation $\vdash_x$ as trivializable if and only if for all α, β, $\{\alpha, \neg\alpha\} \vdash_x \beta$. It is straightforward to show $\vdash_c$ is not trivializable. Furthermore, it is clear that if α is a classical tautology then $\vdash_c \alpha$ holds. It is also straightforward to show that the relation $\vdash_c$ is monotonic, and that it is reflexive for inconsistent inferences. Though it is not in general reflexive, as shown by the example, $\{\alpha \wedge \neg\alpha\} \nvdash_c \alpha \wedge \neg\alpha$. Also, cut, as defined below, fails to hold in general,

$$\frac{\Delta \vdash_c \alpha \qquad \Delta \cup \{\alpha\} \vdash_c \beta}{\Delta \vdash_c \beta}$$

For a counterexample, consider the database $\{\{1\} : \delta, \{2\} : \neg\delta, \{3\} : \delta \rightarrow \alpha, \{4\} : \neg\delta \rightarrow (\alpha \rightarrow \beta)\}$. It is of interest to consider intuitionistic relevant logic (Tennant 1987) where disjunctive syllogism and or introduction hold, but cut fails.

For the following equivalence between RAc logic and classical logic, we define the map 'unlabel' from sets of labelled formulae to sets of (unlabelled) formulae,

$$\text{unlabel}(\Delta) = \{\alpha \mid \text{i}:\alpha \in \Delta\}$$

Result 1 *For all sets of labelled formulae Δ, and all formulae α, the following holds,*

$$\Delta \vdash_c \alpha \text{ iff } \exists \Gamma \subseteq \textit{unlabel}(\Delta) \text{ such that } \Gamma \not\vdash \bot, \text{ and } \Gamma \vdash \alpha$$

This result shows that the $\vdash_c$ consequence relation is the existential consequence relation defined by Resher and Manor (1970).

However, RAc logic also limits deriving inferences such as $\alpha \wedge \neg\alpha$ that a user may need to obtain from $\{\{1\}:\alpha, \{2\}:\neg\alpha\}$. Hence, RAc logic is perhaps too constrained. The logic allows reasoning with consistent subsets of data but does not allow any reasoning with inconsistent subsets. In this way it is stronger than paraconsistent logics for consistent subsets but weaker than paraconsistent logics for inconsistent subsets. But since it seems reasonable to allow some form of paraconsistent reasoning with the inconsistent subsets we address this in the following section.

3 Further logics of restricted access

To address the restriction on reasoning with inconsistent subsets of data, we can increase the strength of our logic by defining paraconsistent proof rules to operate over inconsistent premises. For this we consider the C_ω paraconsistent logic (da Costa 1974), and its natural deduction presentation NC_ω (Raggio 1978). We amend the proof rules to incorporate a function g. Different definitions for g give us different RA logics.

$$p\wedge\text{I} \; \frac{\text{i}:\alpha, \text{j}:\beta, \;\; \text{g}(\text{i}:\alpha) = \text{g}(\text{j}:\beta) = 1}{\text{i}\cup\text{j}:\alpha \wedge \beta}$$

$$p\wedge\text{E1} \; \frac{\text{i}:\alpha \wedge \beta, \;\; \text{g}(\text{i}:\alpha \wedge \beta) = 1}{\text{i}:\alpha} \qquad p\wedge\text{E2} \; \frac{\text{i}:\alpha \wedge \beta, \;\; \text{g}(\text{i}:\alpha \wedge \beta) = 1}{\text{i}:\beta}$$

$$p \rightarrow \text{I} \quad \frac{\begin{array}{c}\emptyset{:}\alpha \\ \vdots \\ \text{i}{:}\beta,\ \text{g}(\text{i}{:}\beta) = 1\end{array}}{\text{i}{:}\alpha \rightarrow \beta} \qquad p \rightarrow \text{E} \quad \frac{\text{i}{:}\alpha,\ \text{j}{:}\alpha \rightarrow \beta,\ \text{g}(\text{i}{:}\alpha) = \text{g}(\text{i}{:}\alpha \rightarrow \beta) = 1}{\text{i} \cup \text{j}{:}\beta}$$

$$p\neg\neg\text{E} \quad \frac{\text{i: } \neg\neg\alpha,\ \text{g}(\text{i:}\neg\neg\alpha) = 1}{\text{i: } \alpha} \qquad \text{p-LEM} \quad \frac{}{\emptyset : \alpha \vee \neg\alpha}$$

We define the consequence relation $\vdash_p$ for RAp logic as follows, where we assume all items in Δ are uniquely labelled with singleton sets.

$\Delta \vdash_p \alpha$ iff $\exists$ i such that there is a proof of i:α
from Δ using the RAp proof rules,
where for all j: β, g(j:β) = 1

Letting g(j:β) = 1 for all j: β, is essentially no restriction on the proof rules. We define the consequence relation $\vdash_{pc}$ for RApc logic as follows, where we assume all items in Δ are uniquely labelled with singleton sets of markers.

$$\Delta \vdash_{pc} \alpha \text{ iff } \Delta \vdash_c \alpha \text{ or } \Delta \vdash_p \alpha$$

This provides a weak merging of the RAc and RAp logics. Essentially, the logics "operate in parallel" without any inferences from the one logic being used by the other logic. We now consider a stronger merging of these logics, where inferences from one logic can be used by the other.

First consider a logic, RAm, where the consequence relation is the union of the RAc and the RAp proof rules as follows,

$\Delta \vdash_m \alpha$ iff $\exists$ i such that there is a proof of i:α
from Δ using the RAc and RAp proof rules
where g(j:β) = 1 iff (1) j:$\beta \in \Delta$
or (2) there is a proof of j: β from Δ
using the RAc proof rules and j $\neq \emptyset$.

For the definition of the function g, part(2) incorporates the condition j $\neq \emptyset$. This is to prohibit classical tautologies to be manipulated by the RAp proof rules. For example, for the database $\{\{1\}{:}\alpha \wedge \neg\alpha\}$, and the classical tautology $\emptyset{:}\alpha \wedge \neg\alpha \rightarrow \beta$, which can be derived by the RAc proof rules, we need to prohibit the application of $p \rightarrow$E.

Note for the definition of $\vdash_m$, we do not worry about RAp inferences being used by the RAc rules, since if the premises for the RAp inferences are inconsistent, then the RAc rules can not use the inferences, and if the premises for the RAp inference are consistent, then RAc rules derive the same inference.

Hence RAm uses the RAc proof rules to generate inferences from the consistent subsets $\Gamma_1, \cdots, \Gamma_n$ of Δ, and then from combinations of these inferences, uses the RAp proof rules to generate further inference, even though $\Gamma_1 \cup \cdots \cup \Gamma_n$ is not necessarily consistent.

Suprisingly, RAm is trivializable. As an example, consider the following database $\Delta = \{\{1\} : \alpha, \{2\} : \neg\alpha\}$. From the classically consistent subset $\{\{1\} : \alpha\}$, we can derive $\{1\} : \beta \rightarrow \alpha$, and then $\{1\} : \neg\alpha \rightarrow \neg\beta$, using the RAc rules. This inference can then be used together with $\{2\} : \neg\alpha$ by the RAp rule $\rightarrow$E to give $\neg\beta$. Hence even though neither RAp nor RAc are trivializable, the merging of them is. Essentially, these logics co-operate in re-introducing EFQ. RAc uses part of a minimally inconsistent subset of the data, to introduce trivial inferences into a implicational formula, and RAp uses the remainder of that inconsistent subset to eliminate the implication, and thereby allowing the trivialization.

To merge RAc and RAp logics without forming a trivializable logic, we need to restrict the RAc inferences that can be used by RAp logic. This restriction is captured by a new definition for the g function used in the RAp proof rules. For this we require the set of literals Φ, where if α is an atom, then $\{\alpha, \neg\alpha, \neg\neg\alpha, \neg\neg\neg\alpha, ...\} \subseteq \Phi$.

We now consider a new logic RAm2, where the consequence relation $\vdash_{m2}$ is defined as follows,

$\Delta \vdash_{m2} \alpha$ iff $\exists$ i there is a proof of i:α
from Δ using the RAc and RAp proof rules
where g(j:β) = 1 iff (1) j:$\beta \in \Delta$
or (2) there is a proof of j: β from Δ
using the RAc proof rules and $\beta \in \Phi$

This gives a stronger form of reasoning than RApc since classical consequences of (consistent parts of) the database, that are in Φ, can used by the paraconsistent relation. So for example from $\Delta = \{\{1\} : \alpha, \{2\}\neg\alpha, \{3\} : \neg\alpha \wedge (\neg\neg\alpha \rightarrow \beta)\}$, we can derive $\{1\} : \neg\neg\alpha$ by RAc, and therefore β by the RAp proof rules.

Result 2 *The consequence relation $\vdash_{m2}$ is not trivializable*

The definition of RAm2 is a cautious development on RApc. Only allowing literals from RAc reasoning to be used by the RAp reasoning is not the closest merging that we could make. For example we could allow more RAc inferences as data for the RAp proof theory. Indeed there is a large space of non-trivializable logics between RApc logic and classical logic. However, to delineate the boundary between trivializable logics and non-trivializable logics in this space, we need to further characterize the way that two logics can co-operate to bring about trivialization.

4 Properties of RA logics

We provide some results that inter-relate the different RA logics via their consequence closures. For any Δ, let $Cn(\Delta) = \{\alpha \mid \Delta \vdash \alpha\}$, and $Cx(\Delta) = \{\alpha \mid \Delta \vdash_x \alpha\}$, where x $\in$ { p, c, pc, m, m2 }. We also require the function L, where if X is a set of labelled formulae, then $L(X) = \{$ i:$\alpha \mid$ i:$\alpha \in$ X and $\alpha \in \Phi\}$.

Result 3 *For all sets of labelled formulae Δ, the following holds,*

$$Cpc(\Delta)=Cp(\Delta)\cup Cc(\Delta)$$
$$Cm(\Delta)=Cp(Cc(\Delta))=Cn(\Delta)$$
$$Cm2(\Delta)=Cp(L(Cc(\Delta))\cup\Delta)\cup Cc(\Delta))$$

Result 4 *For all sets of labelled formulae Δ, the following holds,*

$$\text{If } \Delta\not\vdash\bot, \text{ then } Cp(\Delta)\subseteq Cc(\Delta)=Cpc(\Delta)=Cm2(\Delta)=Cm(\Delta)=Cn(\Delta)$$

Result 5 *For all sets of labelled formulae Δ, the following holds,*

$$\begin{aligned}\text{If } \Delta\vdash\bot, \text{ then } & (i)\ Cp(\Delta)\not\subseteq Cc(\Delta)\\ & \text{and } (ii)\ Cc(\Delta)\not\subseteq Cp(\Delta)\\ & \text{and } (iii)\ Cc(\Delta)\subseteq Cpc(\Delta)\subseteq Cm2(\Delta)\subseteq Cn(\Delta)\\ & \text{and } (iv)\ Cp(\Delta)\subseteq Cpc(\Delta)\subseteq Cm2(\Delta)\subseteq Cn(\Delta)\end{aligned}$$

The results above also raise the question of whether there are other restricted access logics that allow even more inferences to be derived from inconsistent subset of the data. For example, from $\{\{1\}:\alpha\wedge\neg\alpha\wedge\beta,\ \{2\}:\neg\neg\beta\rightarrow\delta\}$, it can be argued that we should be able to derive β, and then $\neg\neg\beta$, to finally get δ. However, none of the RA logics defined in this paper support this. Nevertheless it is clear that we have a number of options in how we define the propagation of the labels, and of the nature of the consistency checks, in the proof theory. Furthermore, we could adopt an alternative paraconsistent logic such as PI^s (Batens 1980) or FR (Anderson 1975) to reason with the inconsistent subsets of the data.

5 Discussion

The RA logics as proposed here allow for the derivation of 'possible' conclusions to be drawn from a database. However, they do not solve many of the wider problems of handling inconsistent information - such as locating inconsistency, resolving inconsistency, or removing inconsistency. Nevertheless the RA logics do offer a non-trivializable logic that could be incorporated in a wider inconsistency system, such as proposed in Gabbay and Hunter (1991, 1992), and Finkelstein et al (1993), and hence form the basis of systems that could address locating, resolving and removing inconsistency.

One of the drawbacks of this approach is the decoupling of the notion of object-level implication, ie. $\rightarrow$, and from the notion of meta-level consequence as captured by the consequence relation. So for example, in these logics we have $\alpha\wedge\neg\alpha\rightarrow\beta$ as a tautology, but in none do we allow β as a consequence of $\alpha\wedge\neg\alpha$ being in the database. Another significant drawback is complexity. Though, this may be ameliorated by approximation techniques. For example checking whether a propositional formula entails a clause is computationally intractable, yet there is a polynomial approaximation that is based on two sequences of entailment relation (Cadoli 1991). The first sequence is sound and not complete, and the second sequence is complete but not sound. Both sequences converge to classical logic.

Finally, there is the relationship with truth maintenance systems that needs to be considered. Of particular interest is the version defined by Martins and Shapiro (1988) that is based on the relevance logic FR of Anderson and Belnap (1975).

6 Acknowledgements

This work has been partly supported by the ESPRIT BRA DRUMS2 project. The first author is SERC senior Research Fellow. Thanks for much helpful feedback are due to Morten Elvang.

7 References

Anderson A and Belnap N (1975) Entailment: The Logic of Relevance and Necessity, Princeton University Press

Besnard P (1991) Paraconsistent logic approach to knowledge representation, in de Glas M, and Gabbay D, Proceedings of the First World Conference on Fundamentals of Artificial Intelligence. Angkor

Batens D (1980) Paraconsistent extensional propositional logics, Logique et Analyse, 90-91, 195-234

Cadoli M and Schaerf M (1991) Approximate entailment, in Trends in Artificial Intelligence, Lecture Notes in Computer Science, 549, Springer

da Costa N C (1974) On the theory of inconsistent formal systems, Notre Dame Journal of Formal Logic, 15, 497 - 510

Finkelstein A, Gabbay D, Hunter A, Kramer J, and Nuseibeh B (1993) Inconsistency handling in multi-perspective specifications, in Proceedings of the Fourth European Software Engineering Conference, Lecture Notes in Computer Science, Springer

Gabbay D (1991) Labelled Deductive System, Technical Report, Centrum fur Informations und Sprachverarbeitung, Universitat Munchen

Gabbay D and Hunter A (1991) Making inconsistency respectable, Part 1, in Jorrand Ph. and Keleman J, Fundamentals of Artificial Intelligence Research, Lecture Notes in Artificial Intelligence, 535, Springer

Gabbay D and Hunter A (1992) Making inconsistency respectable, Part 2, in Proceedings of ECSQARU'93, Lecture Notes in Computer Science, Springer

Martins J and Shapiro S (1988) A model of belief revision, Artificial Intelligence, 35, 25 - 80

Raggio A (1978) in Arrunda A, da Costa N C, and Chuaqui R, Mathematical Logic, Proceedings of the First Brazilian Conference, Marcel Defabier

Resher N and Manor R (1970) On inference from inconsistent premises, Theory and Decision, 1, 179 - 219

Tennant N (1987) Natural deduction and sequent calculus for intuitionisitc relevant logic, Journal of Symbolic Logic, 52, 665-680