CONTENTS

PHILIPPE BESNARD AND ANTHONY HUNTER

INTRODUCTION TO ACTUAL AND POTENTIAL CONTRADICTIONS

This volume of the *Handbook of the Defeasible Reasoning and Uncertainty Management* is about inconsistency and contradictions in logics. In order to provide a context for the volume, we briefly consider in this chapter some general issues of representing and reasoning with notions of contradiction and inconsistency. For the sake of generality in this chapter, logics are here understood as formal systems consisting of a language $\mathcal{L}$ (in the form of a set of formulas) on which an inference operation C is defined. We will only consider syntactical accounts of inconsistency in the sequel. For a semantical approach to accounting for inconsistency, the reader could consider [Grant 1978].

1 EXPRESSING CONTRADICTIONS IN A FORMAL LOGIC

A common approach, to be referred to as the *C-scheme*, is to relate contradictions to inference, stating that inconsistency arises when all formulas are inferred. This can be represented as follows.

$C(T) = \mathcal{L}$ iff T is inconsistent

Unfortunately, in this approach all contradictions are then the same: When considering sets of formulas S closed under C (i.e., $C(S) = S$), then any two inconsistent such sets are the same: $S_1 = C(S_1) = \mathcal{L}$ and $S_2 = C(S_2) = \mathcal{L}$. Yet a fundamental principle of logics is that theories are characterized by the set of formulas that can be inferred.

On the more positive side, this formalization of contradictions involves no assumptions on the vocabulary of the language, not even on the structure of the language (that is, formulas can just be arbitrary items). For instance, the logic language (namely the set of all formulas) could simply be the following set of chains of symbols: $\mathcal{L} = \{rain, snow, sun\}$. For this language $\mathcal{L}$ we could define the following relation C among others: $C(T) = \mathcal{L}$ iff $\{rain, snow\} \subseteq T$. In all other cases, $C(T)$ would simply be T itself. The consistent theories are then the ones that exclude $rain$ or $snow$. The set $\{rain, sun\}$ is such a theory, of which the conclusions drawn by means of C are $rain$ and sun.

A related approach, to be referred to as the *A-scheme*, is to pick a subset of the language, and use each element of the subset as a representation of absurdity. As an extreme, but common case, the selected subset can be a singleton set whose single element is the absurd formula. Now, in a logic where not all absurd formulas yield the same inferences, distinguishing between contradictions is possible, depending on what absurd formulas are involved.

Consider the following example of a logic dealing with two simple shapes, circle and square, and two sizes, small and large, as follows. The language is $\mathcal{L} = \{circle, square, large, small, roundsquare, smallarge\}$. In this, there are two absurd formulas, namely $roundsquare$ and $smallarge$, corresponding to the impossible shape of a rounded square and the impossible size of an object being both large and small. The inferences are then defined as follows.

$$C(T) \supseteq T \cup \{circle, square, roundsquare\}$$
$$\text{if } \{circle, square\} \subseteq T \text{ or } roundsquare \in T$$

$$C(T) \supseteq T \cup \{small, large, smallarge\}$$
$$\text{if } \{small, large\} \subseteq T \text{ or } smallarge \in T$$

$$C(T) = T \text{ otherwise.}$$

Now, contradictions about shapes (e.g., $\{circle, square, small\}$) can be distinguished from contradictions about sizes (e.g., $\{circle, large, small\}$) because the former ones yield $roundsquare$ as a conclusion but not necessarily $smallarge$ whereas the latter ones yield $smallarge$ as a conclusion but not necessarily $roundsquare$.

The A-scheme is similar to the C-scheme in that it is a formalization of contradictions which makes no assumption about the language: The illustration about shapes and sizes shows that the language can be fully unstructured. It is obvious that the formulas cannot be decomposed since they contain no connective, and indeed they can be regarded as arbitrary strings of symbols.

Absurd formulas can sometimes be found in the C-scheme as well: A is an absurd formula within the C-scheme if $C(T) = \mathcal{L}$ for all T that contain A. Given C, any two absurd formulas A_1 and A_2 share the same set of conclusions: $C(A_1) = C(A_2) = \mathcal{L}$. By contrast, the A-scheme admits different absurd formulas that can each have a different set of conclusions: It is not the case that for all T containing any absurd formula then $C(T) = \mathcal{L}$ (otherwise the C-scheme would result).

Curry [1963] calls absurdity the property of a formula implying every formula. A reader familiar with Curry's terminology will have by now noted that absurdity as understood here has a less technical meaning than in Curry's work.

Another point of view, to be referred to as the *N-scheme*, is the one that is followed by all contributions in this volume: Contradictions are captured through an auxiliary notion, that of negation. This introduces an assumption about the language, since we need to incorporate a connective, denoted $\neg$, for negation. So, if A is a formula, then $\neg A$ is also formula, denoting the statement denying the statement represented by A. Hence, $\{A, \neg A\}$ (or $A \wedge \neg A$ if conjunction is available, too) is a syntactical account for inconsistency: Whatever its underlying inference relation, any logic that adheres to the N-scheme includes $\{A, \neg A\}$ (or $A \wedge \neg A$) as a contradiction.

Interestingly, the N-scheme is compatible with either of the C-scheme or the A-scheme. These issues are touched upon in [Curry 1963] whereas [Gabbay 1988] (see also [Gabbay and Wansing 1995, Wansing 1996]) offers a more systematic study mostly focusing on the connection of the A-scheme with negation (an early contribution along these lines being [Nelson 1949]).

An example of combining the C-scheme together with the N-scheme is the following. The language is $\mathcal{L} = \{coloured, \neg coloured, solid, \neg solid, other\}$ (here the language is purposely not closed under the negation connective). The logic is such that $C(T) = \mathcal{L}$ if either $\{coloured, \neg coloured\} \subseteq T$ or $\{solid, \neg solid\} \subseteq T$. Otherwise, $C(T) = T$.

An example of combining the A-scheme together with the N-scheme is the following. The language is $\mathcal{L} = \{coloured, \neg coloured, solid, \neg solid, fool_c, fool_s\}$. The logic could be such that $C(\{coloured, \neg coloured\}) = \{coloured, \neg coloured, fool_c\}$ and $C(\{solid, \neg solid\}) = \{solid, \neg solid, fool_s\}$. The reader can fill in the details as required.

Finally, it is possible to satisfy all three schemes at once as exemplified by propositional logic (in as much as a propositional symbol $\perp$ is available) that verifies the N-scheme, the C-scheme and also the A-scheme in its variant of a singleton set of absurd formulas.

2 INFERRING FROM CONTRADICTIONS IN A FORMAL LOGIC

After deciding on a way of expressing contradiction comes the question of what reasoning should be allowed with contradictions. In other words, when a set of premises involves a contradiction, what constitutes acceptable or rational reasoning? The logics discussed in this volume follow one of two views for information that is in some sense inconsistent. The first view, called the *actual-contradictions view*, is one of tolerating inconsistency, by allowing some reasoning with a set of inconsistent statements, but not allowing the set of inferences to be equal to the language. In other words, the actual contradictions view does not allow trivialization of the reasoning. The second view, called the *potential-contradictions view*, assumes some mechanism to resolve conflicts, thereby obviating the potential of any contradiction.

The actual-contradictions view assumes that no "degenerate" reasoning should occur when contradictory statements are jointly asserted. Furthermore, asserted premises should be fully available for use in deriving inferences. No premise is viewed as weaker than another. In particular, none is discarded as irrelevant or wrong, and they all can be inferences in their own right. An illustration is given by relevant logics [Anderson and Belnap 1975] where $\{A, \neg A, A \wedge \neg A\} \subseteq C(\{A, \neg A\})$ but $B \notin C(\{A, \neg A\})$.

Indeed, in the actual-contradictions view, coping with contradictions goes beyond expressing them. It has to do with the way contradictions are regarded by the user of the logic. A rather classical philosophy is that they are "bad". In other words, they should not occur and if they do, then it is a disaster (basically meaning that the reasoning collapses, and the resulting inferences regarded as trivial). Another philosophy is that contradictions arise, and that systematic eradication of contradictions is a mistake on several grounds. One is a pragmatic reason, contradictions are hard to detect and even harder to remedy to. An even more fundamental objection to the programme of eradicating contradictions is that they exist in their own right and that they can be more informative than any consistent revision of the theory. So, remedying contradictions could result in the loss of information and impoverishment of reasoning. A discussion along these lines can be found in [Gabbay and Hunter 1991] which argues against requiring consistency at any cost.

By contrast, the potential-contradictions view is sympathetic with the philosophy that contradictions do not actually exist. The situation where contradictory statements are asserted then gives rise to two possibilities. One possibility is that some of statements involved in the contradiction can be found "responsible" for the contradiction (of course, "responsible for the contradiction" means much more than simply the fact that without the statement at hand there would be no contradiction: it is rather obvious that such a notion would be symmetrical, hence useless). The other possibility is that the statements are indicative of conflicting arguments whose relative value is clear. In other words, at least one of the arguments involved is known not to stand in the face of another of the arguments involved. In this way, one may foresee such a class of contradictory statements and that the resulting conflict yields a solution where contradictions vanish. Reasoning then conforms to trying to identify the "strongest argument" where all arguments that oppose it are discarded. If no strongest argument can be found, then all competing arguments are discarded.

The potential-contradictions view becomes concrete in the field of so-called defeasible reasoning. An example goes like this.

Typically, birds can fly.

Kiwis cannot fly.

Kiwis are birds.

For this, we can see there is the potential for inconsistency when considering the case of kiwis. Two arguments can be constructed concerning kiwis that have mutually conflicting conclusions. One says that kiwis cannot fly (a trivial argument that coincides with a premise). Another says that kiwis can fly (an argument coming from the statement that kiwis are birds and that birds can fly, typically). Even though inconsistency threatens, it does not actually happen because the first argument rules out the other. Stated otherwise, there is a potential contradiction but it does not turn into an actual contradiction (clearly, kiwis are not typical birds).

As another illustration, the modal formulation [Konolige 1988] of default logic [Reiter 1980] encodes the above example as the following theory T, where $\diamond$ is a modal operator denoting possibility:

$$(bird \wedge \diamond fly) \rightarrow fly$$

$$kiwi \rightarrow \neg fly$$

$$kiwi \rightarrow bird$$

$$kiwi$$

The crucial point is that the problem about the ability, or inability, of kiwis to fly is taken into account: The formalization of the example depends on having represented in a very different way the fact that birds can fly and the fact that kiwis cannot fly. As a result, there is no actual contradiction $fly \wedge \neg fly$ but only the weaker $\neg fly \wedge (\diamond fly \rightarrow fly)$. In symbols, $fly \notin C(T)$. In default logic, actual contradictions are undesirable: $C(\{fly \wedge \neg fly\}) = \mathcal{L}$.

3 LOGICS FOR REASONING WITH CONTRADICTIONS

The following chapters in the volume are mainly concerned with particular examples of logics that can deal with inconsistency. These can be considered in terms of the N-scheme and of the actual-contradictions, and potential-contradictions, views. Essentially, Chapters 2 and 3 are in accordance with the actual-contradictions view, whereas Chapters 4, 5 and 6, are in accordance with the potential-c ontradictions view.

In Chapter 2, Hunter looks at various ways that classical logic can be amended for reasoning with contradictory premises. He considers four cases. In the first case, we can consider a fragment of classical proof theory that is sufficiently weakened so that an arbitrary B fails to be inferred from $A \wedge \neg A$. This is of course a syntactical-based approach, and it covers a number of the more well-known paraconsistent logics. The second case is a semantical-based approach. Here the idea is that conjoining the truth values *false* and *true* can be meaningful and results in a new truth value, namely *contradictory*. This discussion focusses on Belnap's multiple-valued logic [Belnap 1977]. In the third case, we can restrict the use of certain classical proof rules so as to avoid the proof of an arbitrary B following from $A \wedge \neg A$. This is close in spirit to the first case. Finally, in the fourth case, Hunter discusses a simple idea that has been investigated by many people: If the premises form a contradictory collection of statements, then focus on consistent parts of this collection and reason only from such consistent parts. This gives rise to a range of possible systems.

Whenever negation is available and inconsistency is characterized syntactically in the form $A \wedge \neg A$ (the N-scheme), it is still possible to use the C-scheme – we have $C(T)$ if $A \wedge \neg A$ is in T. In such a case, modal languages can be used to cope with the conflicting statements. For instance, the modal operator $\triangle$ can govern a formula A and its denial $\neg A$, in the formula $\triangle A \wedge \triangle \neg A$, without all formulas being inferable from it. In Chapter 3 of this volume, van der Hoek and Meyer provide a review of the use of this technical device from doxastic and epistemic perspectives, where $\triangle$ is interpreted as belief. They also consider a version dealing with multiple agents so that contradictions span over two agents as follows: $\triangle_1 A \wedge \triangle_2 \neg A$ where each $\triangle_i$ operator denotes belief for an agent i. They go on to describe their own application of the above idea to default reasoning via the so-called Epistemic Default Logic. Lastly, they discuss another method where $\triangle_i A$ means that there are at least i possible worlds (in the sense of Kripke) where A holds. Of course, all this again amounts to having $\triangle_1 A \wedge \triangle_2 \neg A$ consistent.

The next three chapters are devoted to the topic of defeasible reasoning. In this kind of reasoning, the potential for contradiction can be anticipated, and the corresponding notion of inference developed to avoid contradictions actually taking place. Technically, whenever there are arguments for both A and $\neg A$, these arguments are vetted so that at most one of them is inferred. For instance, default reasoning makes it possible to cope with statements of the form: from A, conclude B unless inconsistency arises. An illustration is given above, in the form of the kiwis example. The statement about birds being able to fly is amenable to representation according to default reasoning: from $bird$, conclude fly unless inconsistency arises (which is the case when $kiwi$ is asserted together with a formula stating that no kiwis can fly). In Chapter 4, Schaub provides a survey of Default Logic that was invented to capture default reasoning as just presented. In Chapter 5, Delgrande investigates the same idea but using a conditional connective that is written $\Rightarrow$. In Chapter 6, Geerts, Laenens and Vermeir look at related systems called defeasible logics.

In Chapter 4, Schaub first considers the original version of default logic as defined by [Reiter 1980]. From a set of default rules together with a set of classical formulas, jointly called a default theory, default logic works by identifying a mutually acceptable set of inferences that can follow from the default theory. This set of inferences is a set of classical formulas, closed under classical consequence, and is called an extension. If there are conflicting arguments that follow from the default theory, then there may be more than one extension. To avoid contradictions in an extension, each member of the extension is checked for consistency with the other members of the extension. For example, consider the following default rule, where A, B and C are classical formulas.

$$\frac{A \ : \ B}{C}$$

Essentially, if A is in the extension, and B is consistent with the extension, then C is in the extension. Variants of default logic are based on different approaches to the consistency checking. For example, consider a pair of default rule of the following form.

$$\frac{: \ B}{C} \quad \frac{: \ \neg B}{D}$$

For these default rules, using the original version of default logic, we obtain an extension containing both C and D. However, for C, we checked that B is consistent with the extension, and for D, we checked $\neg B$. Yet, $B \wedge \neg B$ is inconsistent. Schaub considers variants of default logic that address problems with the original version of default logic such as this.

Delgrande discusses, in Chapter 5, systems that distinguish two kinds of conditional statements, using two distinct conditional connectives $\supset$ (the well-known material conditional) and $\Rightarrow$ (the so-called variable conditional, or a weak conditional). The informal meaning of a "defeasible conditional statement" $A \Rightarrow B$ is roughly "in normal circumstances, if A then B". A more formal way of looking at this is to consider worlds in the sense of modal logic and to say that "in the least exceptional worlds in which A is true, B is true, too". Least exceptional is meant to be the same as most normal. This therefore offers logics of normality.

A fundamental feature of $\Rightarrow$ is that $A \Rightarrow B$ can be false even when A is false (the case of the so-called counterfactual conditionals). Also, $\{A, \neg B, A \Rightarrow B\}$ is satisfiable. Still another feature is that $A \Rightarrow B$ fails transitivity because there is no need for $A \Rightarrow C$ to be true when $\{A \Rightarrow B, B \Rightarrow C\}$ is true.

Whilst the kind of information being handled is similar to that of default logic (Chapter 4), and defeasible logic (Chapter 6), the aim is not to provide a robust, general-purpose technique for providing inferences from defeasible information. Rather, it is to provide a semantic account of such information. However, Delgrande does also consider hybrid approaches where a conditional logic is not used as an inference system but simply as a means to represent default conditionals so as to generate adequate theories in default logics.

Geerts, Laenens and Vermeir focus, in Chapter 6, on approaches that resort to some structure imposed on theories, in particular, orderings on sets of formulas. At the level of object-language, they go beyond conditional logic by requiring three conditional connectives. Intuitively, these are used to represent the following kinds of information.

"if A then B, no matter what"	this is written $A \Rightarrow B$
"if A then normally B"	this is written $A \rightarrow B$
"if A then normally do not conclude B"	this is written $A \leadsto B$

In the systems reported by Geerts, *et al.*, the fundamental idea is that arguments from premises can be ordered, based on the ordering of the premises. There are two types of ordering. The first is the ordering over the conditional formulas: A conditional using $\Rightarrow$ is preferred to a conditional using $\rightarrow$, and to a conditional using $\leadsto$. Essentially, the order is used to inhibit drawing some conclusions: Given $A \Rightarrow \neg B$ and $A \rightarrow B$, then A yields $\neg B$ and B is discarded; Similarly, given $A \Rightarrow \neg B$ and $A \leadsto \neg B$, then A yields $\neg B$; Of course, given only $A \rightarrow B$, then A yields B; Though, given only $A \leadsto B$, then A yields neither B nor $\neg B$, it can only defeat other rules. Clearly, such a simple criterion is not enough to found a satisfactory notion of inference for non-monotonic (or defeasible) reasoning. The second kind of ordering is over subsets of the sets of premises. So for example, a rule $A \rightarrow C$ could be preferable to a rule $B \rightarrow \neg C$ because A refers to a more specific context than B. As another example, for rules $A \rightsquigarrow C$, and $B \rightarrow C$, if A refers to a more specific context than B, and A and B are assumed, then neither C nor $\neg C$ are derived. There is a range of techniques for identifying such preference orderings over premises.

To make things clearer, we return to the kiwis example. In defeasible logics, the example could be represented by means of the following formulas, where all the formulas are in the same sub-theory:

$$kiwi \Rightarrow \neg fly$$

$$bird \rightarrow fly$$

$$kiwi \Rightarrow bird$$

$$kiwi$$

Since there is an argument for fly and an argument for $\neg fly$, the ordering over conditional formulas is used to allow $\neg fly$ as an inference, in preference to fly as an inference.

In the final two chapters of this volume, relationships with classical logic and with logic programming are established. In Chapter 7, Lenzen considers necessary conditions for negation operators with a focus on paraconsistent negation. As we move away from classical negation, as is necessary to avoid the problems of *ex falso quodlibet* with inconsistent data, we need to consider more deeply the ramifications of these changes. The analysis by Lenzen therefore is in the same vein as earlier investigations such as by Gabbay into the question of "What is negation?" [Gabbay, 1988].

Logic programming is an important route for making the use of logic viable. With this motivation, paraconsistent semantics — based on paraconsistent logics — have been developed for logic programs. On the one hand, we can view this as paraconsistent logics being applied to extend logic programming. On the other hand, we can view logic programming as being a vehicle for solving real-world inconsistency handling problems where the behaviour of the logic programs are in accordance with a paraconsistent logic.

A number of proposals have been made for incorporating paraconsistent semantics into logic programming. These proposals have built upon the notion of extended logic programming which incorporates an explicit form of non-classical negation (as opposed to the notion of negation-as-failure which is the only negation in commercial logic programming). As a result of the explicit negation, extended logic programs can contain contradictory statements. In Chapter 8, Damásio and Pereira survey a range of proposals for paraconsistent semantics for extended logic programs.

4 CONCLUSIONS

In this introduction, we have aimed to indicate some of the difficulties of representing and reasoning with inconsistent information, and to provide an overview of some of the options discussed in the following chapters. Whilst classical logic has many appealing features for knowledge representation and reasoning applications, it is usually inadequate for inconsistent information. Therefore in order to handle actual-contradictions, or potential-contradictions, we need to amend classical logic. This volume can be considered as a presentation and analysis of a range of such amendments. None of the logics presented is ideal for all applications, but there is some value in each of them.

Philippe Besnard
IRISA, France.

Anthony Hunter
University College, London.

REFERENCES

[Anderson and Belnap, 1975] Anderson A. R. and Belnap N. D. Jr. *Entailment. The Logic of Relevance and Necessity*. Princeton University Press, 1975.

[Belnap, 1977] Belnap N. D. Jr. A Useful Four-Valued Logic. In: *Modern Uses of Multiple-Valued Logic*, Dunn J. M. & Epstein G. (eds), pp. 8–37, Reidel, 1977.

[Curry, 1963] Curry H. B. *Foundations of Mathematical Logic*. McGraw-Hill, New York, 1963.

[Gabbay, 1988] Gabbay D. M. What is Negation in a System? *Logic Colloquium '86*, Elsevier, Amsterdam, pp. 95–112, 1988.

[Gabbay and Hunter, 1991] Gabbay D. M. and Hunter A. Making Inconsistency Respectable. *Fundamentals of Logics in Artificial Intelligence Research*, LNCS 535, pp. 19–32, Springer-Verlag, 1991.

[Gabbay and Wansing, 1995] Gabbay D. M. and Wansing H. Negation in Structured Consequence Relations. In: *Logic, Action, Information*, Fuhrmann A. & Rott H. (eds), pp. 328–350, De Gruyter, 1995.

[Grant, 1978] Grant J. Classifications for Inconsistent Theories. *Notre Dame Journal of Formal Logic* 19 (3), pp. 435–444, 1978.

[Konolige, 1988] Konolige K. On the Relation between Default and Autoepistemic Logic. *Artificial Intelligence Journal* 35, pp. 343–382, 1988.

[Nelson, 1949] Nelson D. Constructible Falsity. *Journal of Symbolic Logic* 14, pp. 16–26, 1949.

[Reiter, 1980] Reiter R. A Logic for Default Reasoning. *Artificial Intelligence Journal* 13, pp. 81–132, 1980.
[Wansing, 1996] Wansing H. *Negation. A Notion in Focus*. De Gruyter, 1996.