A survey of knowledge representation formalisms involving uncertainty

Anthony J. Roy

Room 106,
Department of Computer Science,
Keele University,
Staffordshire,
ST5 5BG, U. K.

work@ant-roy.co.uk

http://www.ant-roy.co.uk/research/

April 6, 2001
Introduction

• Uncertainty present in geographic information.
• A need to represent this uncertainty formally.
• Ways of reasoning with uncertainty, inconsistency and partially justified judgements using formal representations.
• How can we use these techniques for dealing with real world problems.
Non-monotonic Reasoning

- Developed to handle reasoning where partially justified judgements are required.
- Incomplete or uncertain data can force us to require such judgements.
- Differs from classical reasoning since adding information to our set of beliefs does not always result in an increase in the inferences we can make.
- Example: Tweety is a Bird, most birds can fly → Tweety can fly. Tweety is a bird, most birds can fly, Tweety is an Emu → need to retract ‘Tweety can fly’.
Default Logic

- Developed by Reiter.
- Gives rules for making partially justified assumptions.
- Of the form: Given that X is true, and A, B, C are not ruled out, then assume W.
- We use these rules to derive sets of consistent conclusions from the initial set of beliefs - called extensions.

Extensions:

1. No default rules can be consistently applied to infer something not already in the extension.
2. Minimal relative to the last rule. (no arbitrary sentences)
Circumscription

- The idea behind circumscription is the minimization of particular abnormality predicates.
- Unless explicitly stated, entities are assumed to be normal.
- Rules similar to default logic: Tweety is a bird, and Tweety is not abnormal, then Tweety can fly.
- We may have many abnormality predicates, e.g. $ab(fly)$, $ab(nospeech)$ etc.
- Well researched - no fixed point equations to solve.

**But** - circumscribing predicates involves an equation in second-order logic: automated deduction??
Autoepistemic Logic

- We assume that an agent can introspectively reason about its beliefs.
- We add a modal operator $L$ to the logic: $L\phi$ means proposition $\phi$ is believed (is in the database).
- Default rules expressed within the language as modal sentences. e.g. $b \land \neg L \neg f \rightarrow f$.
- Extensions are derived from the beliefs of an agent, along with introspective beliefs (e.g. if $f$ is not in my set of beliefs, then $\neg L f$ will be in any extension).
- Similarly to default logic, solving this is a fixed-point problem.
Conclusion

• As they stand, all three non-monotonic logics above are severely limited in terms of computational complexity.

• However, they all present potentially useful ways of dealing with the addition of new (inconsistent) data.

• Ways of simplifying the theories would therefore be desirable to make them more tractable.
Belief Revision

Belief revision comes from the need to remove some sentences from a set of beliefs in order to make room for ‘better’ beliefs. The following points are generally considered important:

1. Our beliefs should be consistent, before and after revision.
2. The way we represent our beliefs should be the same before and after revision.
3. Information lost during a belief change should be kept minimal.
4. Less important beliefs should be retracted in preference to others.
Belief Sets

- Consequentially closed sets of formulae.
- Represent *implicit* beliefs.
- Revision consists of contraction followed by an expansion.
- Coherence Theory: logical structure is important, not where the beliefs come from.
Belief Bases

- A consistent set of beliefs representing explicit beliefs.
- Belief sets can be generated by taking the consequences of a belief base.
- Revisions either on the base to produce a base, or on the belief set derived from the base.
- Foundations Theory: beliefs should be justified in some way.
Conclusion

• Belief sets have a more sound theoretical grounding.
• Belief bases seem more intuitive.
• Belief bases also seem more computable.
Bilattices and Paraconsistent Logic

• Classical logic cannot deal with inconsistent information in a sensible way.

• Aim of paraconsistent logic: To make sensible inferences from inconsistent data.

• Range from weakening classical proof theory to argumentative logics dealing with consistent subsets.

• The following approach uses a four valued logic to allow us to reason with inconsistency.
The Algebra 4

We have a four valued lattice of truth values:

\[
\begin{array}{cccc}
\text{N} & \text{T} & \text{F} & \text{B} \\
\downarrow & \uparrow t & \downarrow & \uparrow \\
\text{F} & \text{B} & \text{F} & \text{N} \\
\end{array}
\]

This can also be seen as a lattice of the amount of knowledge:

\[
\begin{array}{cccc}
\text{B} & \text{T} & \text{F} & \text{N} \\
\downarrow & \uparrow k & \downarrow & \uparrow \\
\text{B} & \text{B} & \text{F} & \text{N} \\
\end{array}
\]
4-valued Logic

• Use a subset of classical connectives: $\lor$, $\land$, $\neg$ corresponding to operations on the lattice.

• We assign valuations from 4 to the variables of the language, and can then derive the truth value of complex sentences.

• We have entailment, $A \rightarrow B$ when in each valuation the value of $A$ is less than the value of $B$.

• We cannot now derive *everything* from the existence of a contradiction.
Bilattices

This paraconsistent logic can also be applied to larger lattices than simply four values. An example may include using possibilistic values in the following way:
Conclusion

• Allows us to express the fact that we have no information about a proposition or contradictory information.

• Allows us to circumnavigate the logical pitfalls of classical logic when we have contradictory information.

• May be able to use values from possibilistic logic, or fuzzy logic to derive a more expressive paraconsistent logic.
Future Work

• Look at possibilistic logic (preference order, ties with fuzzy logic)

• Test formalisms on real-world applications. (Applications from other WP’s).

• Implement software to apply formalisms to more general cases (WP1.5)

• Computational complexity matters.

• Preference by spatial proximity.
THE END