Against Nonmonotonic Logic

Michael J. O'Donnell
The University of Chicago

nonmonotonic behavior

$\not\Rightarrow$

nonmonotonic consequence

- For AI (and logic), the form in which we describe something is crucial, as well as the accuracy of the description

Suppose that we know:

> typical birds fly
> penguins do not fly
> George is a bird

We sensibly conclude:

> George flies

- "typical" just means default

- The connection between "bird" and "typical bird" is the problem

- I made "typical" explicit to avoid formal contradiction

- Making "typical" implicit doesn't solve the problem

- I postulate the need to perform this sort of reasoning

- My English sentences suggest atomic propositions; they do not convey propositional structure

Suppose that we know:

> typical birds fly
> penguins do not fly
> George is a bird

And then we learn that:

> George is a penguin

~~We sensibly conclude:~~

> ~~George flies~~

Now we conclude:

> George does not fly

- The problem is the *removal* of "George flies"; the new conclusion "George does not fly" is clear

Unconventional properties of formal logic

1. defeasible

2. defaultable

3. error-tolerant

4. nonmonotonic

1. defeasible: assertions may be withdrawn

2. defaultable: absent definite information, assume typical case

3. error-tolerant: limit impact of false, even contradictory, assertions

4. nonmonotonic: the logical consequence relation is not monotonic

- 1–3 are essential, 4 is a wrong turn

Deductive Consequence Relation

$$\mathbf{A} \vdash C$$

Accepting $\mathbf{A}$, we sensibly conclude C

- **A** is a finite set of assertions

- In conventional (classical & constructive) logics, we assert perfectly reliable truth

- In practice, we assert effective knowledge (rational belief)

Monotonic Consequence

$\mathbf{B} \supseteq \mathbf{A} \vdash C$ implies $\mathbf{B} \vdash C$

- More knowledge yields more consequences

- Classical, constructive, and relevant logics all use monotonic consequence

George Demands Nonmonotonic Deduction

$$\left\{ \begin{array}{l} \text{typical birds fly,} \\ \text{penguins do not fly,} \\ \text{George is a bird} \end{array} \right\} \vdash \text{George flies}$$

$$\left\{ \begin{array}{l} \text{typical birds fly,} \\ \text{penguins do not fly,} \\ \text{George is a bird,} \\ \text{George is a penguin} \end{array} \right\} \nvdash \text{George flies}$$

- Adding "George is a penguin" seems to remove "George flies"

Does He Really?

$$\left\{ \begin{array}{l} \text{typical birds fly,} \\ \text{penguins do not fly,} \\ \text{George is a bird,} \\ \text{that's all I know} \end{array} \right\} \vdash \text{George flies}$$

$$\left\{ \begin{array}{l} \text{typical birds fly,} \\ \text{penguins do not fly,} \\ \text{George is a bird,} \\ \text{\sout{that's all I know}} \\ \text{George is a penguin} \end{array} \right\} \nvdash \text{George flies}$$

7

- "that's all I know" has complex structure, refers to previous assertions and conclusion

- more carefully, "an acceptably diligent search reveals no other information relevant to George's capacity for flight"

- removal of "that's all I know", not addition of "George is a penguin", removes "George flies"

- defeasible behavior with monotonic consequence

- defeasibility is a property of assumption gathering, not of deductive consequence

The Alternatives

$\{\ldots$ that's all I know $\ldots\} \quad \vdash_L \quad C$

versus

$\{\ldots\} \quad \vdash_G \quad C$

- $\vdash_L$ (L for "local") is conventional deductive consequence—whenever you know hypotheses, draw conclusion

- $\vdash_G$ (G for "global") is nonmonotonic deductive consequence—from *all* your knowledge, draw conclusion

Properties of Reasoners

1. **R** has access to huge knowledge base

2. Access to knowledge is incrementally costly

3. **R** must always reason from small subsets of knowledge base

4. **R** interleaves inference with search

1. knowledge base includes local storage, communication from other agents, observable real world

2. cost is almost always monotone in number of propositions read

3. cannot scan all knowledge before inferring (follows from 1–2)

4. search may be complex, inference may affect further search

The Real Problems

- defeasibility—remove obsolete knowledge

- search control—find relevant knowledge cost-effectively and invoke defaults

- error tolerance—survive incorrect/inconsistent knowledge

- defeasance is an operation on a knowledge base, complementary to adding knowledge— probably requires more structure than set of assertions—requires some analysis of relevance

- search control—may benefit from same structure that supports defeasibility

- error tolerance (paraconsistency) may come from relevant logics—damage control, not perfect defense