The Knowledge Engineering Review, Vol. 12:4, 1997, 407–409

Logical methods for computational intelligence

GRIGORIS ANTONIOU[1] and NEIL V. MURRAY[2]

[1]*School of Computing & Information Technology, Griffith University, QLD 4111, Australia, email:ga@cit.gu.edu.au*
[2]*Department of Computer Science, University of Albany – SUNY, Albany, NY 1222, USA, email:nvm@cs.albany.edu*

Over the past years, two main approaches to computational intelligence have emerged: the symbolic and the non-symbolic approach. The perhaps most prominent methods of the symbolic approach are based on *logic*. Logical methods exhibit a series of desirable properties:

- Transparent representation of information.
- Precise understanding of the meaning of statements (semantics).
- Sound reasoning methods.
- Explanation capabilities.

A special session on logical methods for computational intelligence was held at the *3rd Joint Conference on Information Sciences*. The field of computational logic is so broad that it is impossible to review the main developments in an article. Therefore, in the following we will restrict attention to two areas that turned out to be the focus of the special session: automated reasoning, and reasoning with incomplete and changing information.

The field of *automated reasoning* seeks to develop automatic, or semi-automatic "theorem proving" programs. The development of such programs also involves (sometimes more general) processes like "inference" or "deduction". The generic problem of automatic theorem proving was summarized elegantly by J. A. Robinson (1967):

> "By a *theorem-proving problem* we mean a problem which has the form: *show that B follows from* $A_1, \ldots, A_n$, where $A_1, \ldots, A_n$ and B are *statements*. It may well be that $A_1, \ldots, A_n$ are the *axioms* or *postulates* of some theory, and that B is a (presumed) *theorem* of that theory; in that case the problem is that of *proving* the given theorem from the given axioms.
>
> It is usually thought that such problems require a certain amount of ingenuity for their solution, and indeed it is by using mathematical skill and ingenuity in reasoning that people usually solve them. However, [under very general conditions], a theorem-proving problem can be solved *automatically* if it can be solved at all. That is to say, if the given B *does* follow from the given $A_1, \ldots, A_n$, then this fact can be detected, automatically, by executing a certain purely clerical algorithm on $A_1, \ldots, A_n$ and B as input".

Of course, developing more efficient and powerful versions of the above mentioned "purely clerical algorithm" is an ongoing research problem—one that must be solved if automated reasoning programs are to find practical application. Much progress has indeed been made in the years since those remarks were made. Recently, a significant milestone has been achieved with the solution of the Robbins problem (McCune, to appear). This is the first problem automatically solved by computer that had remained open for so long (about 60 years) and had been studied by such prominent mathematicians (including Tarski). It can certainly be considered an example of computational (or artificial) intelligence.

Theorem provers have also found extensive application outside the boundaries of artificial intelligence, mainly in the area of program verification and derivation. Much progress has been made here as well, but still there is a disparity between machine-generated proofs, and proofs by humans (as they can commonly be found in mathematics textbooks). Therefore, some researchers are turning attention away from optimizing existing techniques and towards the use of planning techniques, to better model the way in which humans develop proofs (Siekmann, 1996).

Other researchers seek to present machine-generated proofs in a way that appeals to humans (e.g.

Huang, 1994). The idea is to summarize chunks of the proof into "macro-steps" on a more abstract level that better resemble steps humans would take. Fehrer and Horacek presented in our special session some work in this direction. Their paper is concerned with the presentation of inequalities in mathematical proofs. Earlier investigations developed the idea that equalities can be quite naturally represented as a chain structure. Fehrer and Horacek show that the crucial property of equality that enables this natural representation is transitivity, a property shared by inequality. Thus they are able to generalize these ideas to chains of inequalities.

Intelligent systems are often faced with the problem of *reasoning with incomplete information*. That means, the information actually needed to make a guaranteed correct decision is unavailable (or there is not enough time or resources to collect the necessary information), yet the system must make some decision that leads in some sense to adequate performance. Plausible assumptions must be made about the missing pieces of information. Depending on how these assumptions are made and justified, we get different approaches to reasoning with incomplete information.

One way is to use probabilistic information, which might lead to the use of probabilistic reasoning, as in probabilistic logic (Nilsson, 1986) and the Dempster–Shafer theory (Shafer, 1976). *In default reasoning*, assumptions are based on default rules, while in *autoepistemic logic* (Moore, 1985) assumptions are based on introspection.

Default reasoning, and in particular default logic (Reiter, 1980), was a particular focus of the special session we organized. It extends classical logic by default rules. A simple example is the closed world assumption known in the database area for a long time: If an assertion is not found in the database then it is assumed false. For example, if we look up a database of flights and don't find a direct flight from Osnabrueck to Albany, then we assume that there is no such flight. Of course, this is only an assumption, since there might be an extra flight due to a special event, for example.

Implementation of default reasoning is an area that was neglected for quite some time, but recently it has attracted a lot of attention, having led to some powerful systems (e.g. DeReS (Cholewinski et al., 1996)). In our special session, the paper by Antoniou dealt with stratification, a technique for decomposing a knowledge base into parts which can lead to efficient default reasoning.

Increasingly, the focus has switched towards pragmatic and applicational aspects of default reasoning. Indeed, default logic has some representational advantages over classical logic, which makes it suitable for several application domains:

- It works properly with conflicting information. While in classical logic a trivialization occurs in the face of an inconsistency, default logic gives an overview of all maximal possible world views supported by the given knowledge.
- Defaults are a natural way of expressing assumptions, and prevail in many domains. Also the use of defaults may lead to more compact, and thus more easily manageable representations. There are indeed theoretical results (Cadoli et al., 1994) which indicate that default logic representations can result in much more compact representations, compared to classical specification methods.

In our special session, the paper by Ghose outlined the use of default reasoning in Requirements Engineering. This seems to be a particularly fruitful domain: requirements often contradict one another due to the presence of different stake-holders with different needs. A default representation helps identify the sources of conflict and ways of resolving them. This information is useful during the next round of meetings with the stake-holders.

Ghose's paper also deals with evolving and changing requirements, which are a very common phenomenon in the early stages of a software project. *Reasoning with changing information* is a more general problem that needs to be addressed in principled ways. One possibility is to use belief revision techniques (Alchourron et al., 1985). Such techniques have been the focus of much theoretical work in knowledge representation over the past ten years. They are beginning now to find application in different domains, for example in marketing (dealing with changing consumers' preferences).

Finally we mention an important, fundamental question: Where do default rules come from? This question is related to *abduction* and *induction*. At the special session, the paper of Lamma et al. describes a method for learning rules with exceptions, which are a kind of default rules.

Overall, logical reasoning is a lively and rapidly developing area. Selected papers of the conference session will be published in a special issue of *Information Sciences*, and we hope to hold a similar event at the *4th Joint Conference on Information Sciences* in 1998.

References

Alchourron, CE, Gardenfors, P and Makinson, D, 1985. "On the logic of theory change: partial meet contraction and revision functions" *Journal of Symbolic Logic* **50** 510–530.

Cadoli, M, Donini FM and Schaerf, M, 1994. "Is intractability of non-monotonic reasoning a real drawback?" *Proc. 12th National Conference on Artificial Intelligence (AAAI-96)* AAAI Press, 946–951.

Cholewinski, P, Marek, V and Truszczynski, M, 1996. "Default reasoning system DeReS" *Proc. 5th International Conference on Principles of Knowledge Representation and Reasoning* Morgan Kaufmann.

Huang, X, 1994. "Restructuring proofs at the assertion level" *Proc. 12th Conference on Automated Deduction* Springer, LNAI 814, 738–752.

McCune, W, "Solution of the Robbins problem" *Journal of Automated Reasoning"* (to appear).

Moore, R, 1985. "Semantical considerations on nonmonotonic logic" *Artificial Intelligence* **25** 75–94.

Nilsson, N, 1986. "Probabilistic logic" *Artificial Intelligence* **28** 31–56.

Reiter, R, 1986. "A logic for default reasoning" *Artificial Intelligence* **13** 81–132.

Robinson, JA, 1967. "A review of automatic theorem proving" *Annual Symposia in Applied Math. XIX* American Mathematical Society, Providence, 1–18.

Shafer, G, 1976. *A Mathematical Theory of Evidence* Princeton University Press.

Siekmann, J, 1996. Keynote speech, *4th Pacific Rim International Conference on Artificial Intelligence* Cairns.