Concurrent Theorem-Proving as Multi-Agent Computation

Michael Fisher

Logic and Computation Group, Department of Computer Science
University of Liverpool, Liverpool L69 7ZF, U.K.

Email: M.Fisher@csc.liv.ac.uk
WWW [personal]: www.csc.liv.ac.uk/~michael
WWW [group]: www.csc.liv.ac.uk/research/logics

It is clear that, in contemporary multi-agent applications, the complexity of problems being tackled and the variety of heterogeneous agents being used are both increasing rapidly [7]. Agents are required to carry out interactive activities, such as *distributed problem-solving* and *negotiation* in order for the multi-agent system to tackle complex problems. However, there is, as yet, no clear formal model for multi-agent activity of this kind.

Our work in this area concerns the development of a simple concurrent theorem-proving approach for classical logic [3] and its use as *exactly* such a model of multi-agent computation. Hence, we have shown how both distrbuted problem-solving [5] and negotiation [4] can be characterised.

Concurrent Theorem-Proving

In [3], we introduced a novel approach to concurrent theorem-proving based upon the distribution of classical logic clauses amongst autonomous agents and the collective generation of a refutation by passing information between these agents. We showed that not only is this a natural mechanism for agent-based distributed proof, but that the distributed proof algorithm is complete for classical logic. A particular facet of the approach is that, if Horn clauses are considered within agents only positive literals need to be communicated, while if non-Horn formulae occur within the agents, then Horn clauses must be communicated between agents.

Distributed Problem-Solving

The view of problem solving as *theorem proving* has a long and influential history in AI, going back at least to the work of Green [6]. This deductive view of problem solving has been particularly influential and useful in AI planning research. *Distributed* problem solving, wherein a group of decentralised semi-autonomous agents cooperate to solve problems, is perhaps the paradigm example of multi-agent activity, and is certainly the most-studied process in distributed AI [1].

The link between our model of concurrent theorem-proving and computational activity in multi-agent systems led us to use the model in order to represent distributed problem-solving [5]. The particular type of distributed problem-solving considered was distributed planning. Here, each agent shares a portion of the problem (and plan operators) and the agents communicate together in order to collectively generate a solution to the problem. An interesting aspect of this work is how different organisations of agents correspond to different varieties of planning.

Negotiation

Negotiation is a key technique for dynamically organising activity in multi-agent systems, yet the negotiation process itself can often be difficult to reason about. Negotiation is defined as "a general process involving communication between two or more agents that attempts to find a way of mutually satisfying the separate agent's requirements by undertaking appropriate actions". Generally, practical negotiation follows a similar style to negotiation amongst humans, with an initial suggestion being followed by alternative proposals, and critiques being passed between the participants, until (hopefully) an acceptable solution is found.

In [4], we considered how simple forms of negotiation between agents could be characterised as particular types of proof in the above concurrent theorem-proving framework. The key insight is that, once agents are able to communicate Horn clauses to each other, they are able to express that "if someone does X and Y for me, then I will do Z" as a Horn clause such as $(X \wedge Y) \Rightarrow Z$. Here, we view agents as *announcing* proposals (and counter-proposals) rather than sending them to specified target agents, and we show that characterising the agents' negotiating behaviour in this way allows us to decide whether a simple successful negotiation is possible or not.

We also examine how an ordering or weighting strategy, when applied to clauses within an agent, can be viewed as constraining the order in which agents negotiate information. Thus, the agent might have several possible responses to a proposal and decides which response to send first based upon its ordering strategy. If this ordering strategy is extended to cover clauses across *all* agents then this provides a characterisation of a form of selection concerning which agents to negotiate with [4].

Future Work

Our current and future work involves extending the range of multi-agent activities that can be characterised with this simple model of concurrent theorem-proving. Our aim is not only to provide a formal basis for multi-agent activity, but also to develop simple abstractions that will be used as the basis for a comprehensive, logic-based, programming language for multi-agent systems [2].

References

[1] A. H. Bond and L. Gasser, editors. *Readings in Distributed Artificial Intelligence*. Morgan Kaufmann, 1988.

[2] M. Fisher. A Logical Approach to Programming Multi-Agent Systems. (In preparation.).

[3] M. Fisher. An Open Approach to Concurrent Theorem-Proving. In J. Geller, H. Kitano, and C. Suttner, editors, *Parallel Processing for Artificial Intelligence*, volume 3. Elsevier/North-Holland, 1997.

[4] M. Fisher. Characterising Simple Negotiation as Distributed Agent-Based Theorem-Proving. In *Proceedings of the Fifth International Conference on Multi-Agent Systems (ICMAS)*, 2000.

[5] M. Fisher and M. Wooldridge. Distributed Problem Solving as Concurrent Theorem-Proving. In *Multi-Agent Rationality*. Springer-Verlag, 1997. (Lecture Notes in Artificial Intelligence, volume 1237).

[6] C. Green. Application of Theorem Proving to Problem Solving. In *Proceedings of International Joint Conference on AI*, 196.

[7] N. R. Jennings and M. Wooldridge. Applications of agent technology. In *Agent Technology: Foundations, Applications, and Markets*. Springer-Verlag, Heidelberg, 1998.