An Interactive Proof Development Environment + Anticipation = A Mathematical Assistant?

Jörg Siekmann, Helmut Horacek, Michael Kohlhase,
C. Benzmüller, L. Cheikhrouhou, D. Fehrer, A. Fiedler,
S. Hess, K. Konrad, A. Meier, E. Melis, V. Sorge
FB Informatik, Universität des Saarlandes, Germany
http://www.ags.uni-sb.de/

Keywords: Automated Theorem Proving, Proof planning, Mathematical Assistant System, Proof Presentation, Agents.

Summary

The effective use of automated theorem provers is frequently augmented by embedding these systems into interactive proof development environments. However, in order to act as a competent mathematical assistant, which is the ambitious goal behind these interactive extensions of automated provers, these tools behave too passively and in a stereotypic way, because they lack the capability to adequately take into account requirements on proof search control and user demands for their own actions. Motivated by this deficit, we have incorporated several facilities into the ΩMEGA proof development system that anticipate a number of divergent factors, based on mathematical knowledge, proof search defaults, and expectations about users. The techniques enhance the system's functionality through proof planning by knowledge-intensive methods, proof search guidance by default suggesting agents, and proof presentation by redundancy avoidance measures. The system's behavior suggests that anticipation is without doubt a central driving force in a mathematical assistant, but this concept must be interpreted in meaningful ways for a variety of techniques, and its manifestations need to be elaborated in greater detail.

Shortcomings of Interactive Proof Development Environments

Insights in the limitations of fully automated theorem provers have motivated the design of interactive proof development environments. The basic idea is that a mathematician who is developing large proofs should be supported by a computer system to which he can delegate subproblems. In practice, however, it has frequently turned out that this simple distribution of labor is insufficient for rendering this work effective. In our view, this situation is caused by some serious deficits that most current theorem prover interfaces suffer from:

- The level of abstraction of the logical calculus of the theorem prover is inadequate for influencing its proof strategies in a knowledgeable way, in particular, the system needs to restrict the communication to a vocabulary a mathematician is familiar with.

- Control mechanisms for guiding the proof search are widely restricted to static parameters that capture dynamic aspects of the proof state in an insufficient manner and do not allow interactive additions and modifications.

- Facilities for specifying the problem to be proved and for presenting the results obtained are generally inconvenient for users. In particular, directly converting the output of a theorem prover into natural language leads to presentations that users consider to be redundant in large parts.

It is our firm belief that improving this situation requires a system to make better use of communicative resources by exploiting defaults and expectations of various sorts. Doing this means taking potential future actions and states into account in determining its actions, which is *anticipation* in a broad sense.

Manifestations of Anticipation in an Interactive Proof Development Environment

In our research group, we are developing the system ΩMEGA [**?**], which is an interactive, deduction system with the ultimate goal of supporting theorem proving in main-stream mathematics and mathematics education, aiming to extend this system to a mathematical assistant. ΩMEGA's design is motivated by insights about the deficits of current interactive proof development environments. Consequently, anticipation manifests itself in several of its components:

- ΩMEGA's proof planner constructs a proof plan for the *goal node* from a set of *supporting nodes* (the proof assumptions) using a set of proof planning operators, called methods, whose contextual usefulness is judged by *control rules* [**?**]. The control rules associated with a method *anticipate the suitability of this method* for the problem at hand.

- ΩMEGA uses agents with specific proof technique expertise to enhance control over proof search. These agents are part of a multi-layered focusing mechanism for computing suitable default values supporting the application of inference rules in a proof state [**?**]. The agents *anticipate the usefulness of a certain proof strategy* in dependency of properties of the current proof state.

- ΩMEGA uses an extension of the PROVERB system [**?**] that presents proofs in natural language. In order to emphasize concise texts that resemble those

found in mathematical textbooks, PROVERB employs a small number of contextually motivated rules expressing aspects of conversational implicature [?]. These rules *anticipate the addressee's inferences* in adapting the proof output to the user's needs.

At the moment we are incorporating further techniques in which anticipation manifests itself into the ΩMEGA system. They include the exploitation of structured mathematical knowledge, extensions to the planner so that it can run in a reactive mode, and provision of more natural and convenient ways to specify a problem.

Assessing the Role of Anticipation

Once we have examined the role of anticipation in major components of ΩMEGA, we have to address the question raised by the title of this paper, that is, to what extent anticipation is the driving force in promoting an interactive proof development environment into a mathematical assistant. On the one hand, the techniques outlined here demonstrate the diversity of situations and the capabilities to act adequately therein, in which anticipation appears as a central concept. On the other hand, for a system to be truly called a mathematical assistant, many of these techniques need to be elaborated in greater depth and sophistication and complemented by some other techniques such as knowledge representation and reasoning skills of various sorts, in which anticipation plays a subordinate role if at all. Hence, we might conclude that anticipation is without doubt a central concept needed for building a mathematical assistant, but its adequate interpretation and the elaboration of its manifestations require considerable effort.

References

[BCF+97] C. Benzmüller, L. Cheikhrouhou, D. Fehrer, A. Fiedler, X. Huang, M. Kerber, M. Kohlhase, K. Konrad, E. Melis, A. Meier, W. Schaarschmidt, J. Siekmann, and V. Sorge. ΩMEGA: Towards a mathematical assistant. In William McCune, editor, *Proceedings of the 14th Conference on Automated Deduction*, number 1249 in LNAI, pages 252–255, Townsville, Australia, 1997. Springer Verlag.

[BS98] Christoph Benzmüller and Volker Sorge. A Focusing Technique for Guiding Interactive Proofs. Submitted to the 8th International Conference on Artificial Intelligence: Methodology, Systems, Applications, 1998.

[HF97] Xiaorong Huang and Armin Fiedler. Proof verbalization as an application of NLG. In Martha E. Pollack, editor, *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, Nagoya, Japan, 1997. Morgan Kaufmann.

[Hor98] Helmut Horacek. Generating inference-rich discourse through revisions of RST-trees. In *Proceedings of AAAI-98*. Morgan Kaufman, 1998.

[Mel98b] Erica Melis. AI-Techniquese in Proof Planning. In *Proceedings of ECAI-98*. John Wiley & Sons, Ltd., 1998.