# The Value of Automated Deduction

Robert L. Constable
Cornell University

**Abstract**

This note looks at the intrinsic value of the subject of Automated Deduction (AD) and considers the state of the field — its accomplishments, promise and problems. I argue that the subject is profound, one of the most promising of the new sciences of the closing century. The field is healthy with first rate scholarly outlets and professional organizations. The existing applications are paying off in the expected time frame of 17 to 25 years from initial attempts and new areas are opening with equal promise; these will keep the field vital while new basic science continues to deepen it. I think that these are the facts, and they are encouraging.

I also claim that the time has come to build more open AD systems that can cooperate with each other. Such open systems will significantly increase the scope and impact of our work, and they will improve the methodology of the field and help energize it.

## 1 Intrinsic Value of the Subject

### 1.1 Fundamental nature of AD

Deduction is basic to thought, and the idea of automating it or using machines to amplifying our ability to think has fascinated people for decades. The systematic study of this idea in AI has great popular appeal (as astronomy does); this appeal is a major asset that few subjects enjoy.

Deductive thought is basic to science. Part of the study of deduction is the study of knowledge management — its organization and retrieval. Improvements in our ability to manage knowledge and carry out complex deductions will advance all of the sciences and mathematics. This fundamental role of deduction was already recognized by the ancients, and its systematic study in logic has a long history.

The logical discoveries of the 20th century are some of its most profound — including the creation of formal systems; the definition of computability; the Godel, Tarksi, and Church theorems; the axiomatic systems that shape modern mathematics such as ZF set theory and the descendents of Principia Mathematica. These results are the mathematical basis and the illustrious heritage of our subject.

Automated Deduction is distinguished from logic by the role of computers in improving our ability to carry out deductions. The field would not exist without the hardware and software used to implement formal mathematical theories. The combined achievements of Logic and Computer Science have allowed us to dream of the Qed project and to offer a technology to solve some of the hardest problems in the engineering of hardware and software systems.

### 1.2 AD research program

The basic research program of the Automated Deduction community is a very grand enterprise indeed. We aim to design computer systems that will allow us to formalize vast amounts of mathematics, that will allow us to formally specify mathematical theories and engineering problems, that will assist us in creating theorems and theories, that will check our inferences, find counter-models to our conjectures, keep track of logical dependencies in proofs, designs and artifacts, and in the end help us solve problems, including hard open problems in mathematics as well as critical programming problems. All of this depends on the deep results in logic and on the remarkable advances of Computer Science.

## 2 State of the Field

### 2.1 Assessment

The field of Automated Deduction has forged its identity through first-rate outlets for its scholarly and scientific work. There are the CADE, CAV and LICS conferences among others. There is the Journal of Automated Reasoning and the Journal of Symbolic Computation as well as affiliation with the Association for Symbolic Logic. (These ties will be on display at the Federated Logic Conference this July at Rutgers — an effort I am proud to have helped organize when I was general chair of LICS.)

A broad view of the field would reveal an international effort involving perhaps nearly a thousand workers whose interests overlap dozens of other areas of research from geometry and algebra, programming languages and logics to psychology and cognitive science. It would include many distinguished scientist and mathematicians. The field is wider still if we include those whose interests have at one time overlapped with AD and who understand the subject well. This creates a large reservoir of good will toward Automated Deduction.

I think it is important that the field define itself broadly in order to mix new ideas into our research program and attract young talent. One means of attracting this talent is to write about the subject in the most accessible way in the general science literature. Another way is to teach the subject well.

### 2.2 Applications

The software systems created as part of the research program are of practical value. They have helped with mathematical advances and they have created a "verification and checking" technology — in model checking, type checking, and proof checking (including programs, protocols and circuits) which is already valuable to industry. These software systems include the family of logic and constraint programming languages.

## 2.3 Promise

The research program is not only of high quality but has very high potential for even more substantial achievements in the near future. We can expect more basic science. This will include algorithmic advances in rewriting, in model checking, in decision procedures, in programming logics, in higher-order logic programming and in type theory to name a few.

It will include new applications in *symbolic algebra* and scientific computing and in education, especially in geometry. The existing applications will be improved, and more of them will become part of *standard industrial practice*. We can see a time-line of advances stretching out for 10 years. Some steps are extensions of the technology that has worked, some are new applications and some are grand challenge problems, and the decade or more of work toward them will shape the field.

The Automated Deduction community will be called upon by the government and industry to help solve problems of national importance — like the current concern about the security of our computing and communication infrastructure. We now have the technology to help, but is is important that it not be over-sold as it was in the 70's. Hyperbole about the future will only detract from the significance of what we can do now.

## 2.4 Problems

As with any field, there are problems in managing growth and in competing for resources both financial and human. Many other fields of science are very vital, and Automated Deduction must draw a lot of its talent from Computer Science which is a booming field with many successful subareas that draw heavily on the talent pool.

We need good mechanisms for attracting and educating talented people. One way to do this is to cast our net widely. This requires relating well to the important allied fields and their organizations without loosing our identity.

## 2.5 Special problems

The researchers in the field who have built systems, dozens of them too numerous to mention, have special problems because they need users to validate their results and need substantial funding to continue building and experimenting. As the number of good systems grow, this becomes a larger problem. People tend to be protective and defensive about their artifacts. Since it is difficult to build on other people's work, people rebuild many things themselves. This exacerbates the problem.

## 2.6 Solutions

The Automated Deduction community is ready to build more open systems. We see in the PC community that it can be done very well with tools like OLE. There are new symbolic algebra systems like OpenMath that also show promise. At Cornell we have considered an approach called Collaborative Mathematics Environments (www@cs.cornell.edu/projects/nuprl).

I think that it is very important to be able to share libraries of formal mathematics as well as to share algorithms, decision procedures, tactics, and proof plans. This would be possible with new open architectures especially if we adopt some of the software and methods of the PC community and practices of the scientific computing community, e.g. high quality libraries, a tradition of cooperation, agreement on languages.

Building for cooperation will require new theoretical work, new programming practices and new designs. For example Doug Howe has proven that it is possible to find a common semantics between constructive and classical type theories which allows sharing theories. He has also begun to examine how to do this in practice. I think that this is a paradigm for future work.

## References


[1] J. Howe Douglas. Importing mathematics from HOL into Nuprl. In *Proceedings of The 1996 International Conference on Theorem Proving in Higher Order Logics*, 1996. To appear.

[2] Douglas J. Howe. Semantic foundations for embedding HOL in Nuprl. In *Proceedings of AMAST'96*, 1996. To appear.