Looking Ahead * **

Peter B. Andrews

Carnegie Mellon University, Pittsburgh, PA, U.S.A.
andrews@cmu.edu
http://gtps.math.cmu.edu/andrews.html

Abstract. We discuss some of the opportunities and problems which may confront the field of automated reasoning in the years ahead. We focus on various issues related to the development of a Universal Automated Information System for Science and Technology, and the problem of developing institutional support for long-term projects.

1 Introduction

While I wasn't immediately aware of it, the field of automated deduction got started just about when I entered graduate school in the fall of 1959. Actually, a few papers in the field had appeared shortly before that, but there was a burst of activity in 1960 with the appearance of the pioneering papers by Davis and Putnam [9], Gelernter, Hansen, and Loveland [11], Gilmore [12], Prawitz, Prawitz, and Voghera [15], [16], and Hao Wang [28], [29]. These were soon followed by [30], Davis's paper "Eliminating the Irrelevant from Mechanical Proofs" [10] in 1963, and then in 1965 Robinson's paper introducing Resolution [18], which started a flood of activity. So, I think of the field of automated deduction as about 50 to 55 years old.

I can't remember when I first became aware of what was going on in automated deduction. The only historical clue I have is an entry in my journal from February 1961 that I should read Hao Wang's paper "Proving theorems by pattern recognition, Part I" [28]. I was certainly familiar with Paul Gilmore's paper [12] when I met him as I started my summer job with IBM in the summer of 1961.

As I think about what has happened in this field in the past 50 years, I'm overwhelmed by all that has been accomplished. People have had lots of ideas and plans, some of which involved questions or experiments with disappointing answers, some of which had serious flaws, some of which were too ambitious, some of which got dropped for extraneous reasons, and some of which turned out to be really successful.

It's very difficult to predict what will happen in the future, but it's very worthwhile to reflect on factors which may affect what will happen in the future.

* This is the text of an invited lecture presented to IJCAR 2012, the 6th International Joint Conference on Automated Reasoning, in Manchester, United Kingdom, on June 27, 2012, as a Guest Speaker.

Sometimes it seems that nothing significant is happening, but when you compare the present with the world fifty years ago, you realize that the future really does come. So let's give a little thought to some things that might affect the future of automated deduction.

2 The Universal Information System

One of the amazing phenomena of the past few decades has been the development of the world wide web as an information source. While it's really a collection of many information sources, those searching for information on the web today tend to regard "the web" as one monolithic information source. As more and more information is made available on the web, it is evolving into a Universal Information System.

One fundamental improvement which we can anticipate for this system is the incorporation of facilities for making logical inferences automatically. When one uses an information system, one may be seeking information which is not explicitly in the system, but can be derived from information which is there. Thus, it is desirable that information systems be organized so that one can apply logical inferences to them. This is particularly important if it is a computer, rather than a person, that is looking for the information. People often make inferences almost subconsciously, but reasoning by computers must be explicit. As the system becomes increasingly effective at answering questions, in particular from the realms of mathematics, science, applied science, and technology, where logic can readily be applied, the Universal Information System will evolve into a Universal Automated Information System.

What will the Universal Automated Information System of the future be like? One possibility is that it will be a huge kluge of systems designed by specialists from various fields using whatever techniques and programming methodologies they found most immediately convenient. Another possibility is that it will be a well organized union of information systems in which formal logic and techniques of automated deduction play significant roles.

A system having all the desirable features we can envision may take centuries to develop, but the Automated Information System of the future is growing right now, and it is important to encourage growth in the right directions. Good plans about how such a system should grow could have enormous future benefits. Let us think about the design and development of a Universal Automated Information System for Science and Technology.

Of course, many people have been thinking about such a system, in at least a casual way, for many years. Indeed, the QED project [17] is concerned with building such a system for mathematics, and many of the same considerations apply to both.

3 The need for modularity

The Automated Information System will grow continually as new knowledge is made available to it. Undoubtedly it will have many components which have been contributed by many different people and projects at many different times. It seems desirable that the content of information in the system be separated from mechanisms for retrieving and deriving information, so that improvements in the retrieval and inference mechanisms can be made in a modular way which will benefit users of all components of the system.

4 Many different disciplines

Sometimes one asks a question which can be answered only by using knowledge from several disciplines. Thus, it is highly desirable that the Automated Information System cover all branches of technical knowledge, and be well integrated to facilitate dealing with such questions.

5 The need for formal logical languages

Most information is written in a natural language (such as English or French or Chinese), but to obtain the benefits of automated deduction, much of the information should be represented in a form which can at least be readily translated into a formal language for which there are well defined rules of logical inference, i.e., a formal logical language.

Thus, there is a need for formal logical systems, which are formal languages with rules of reasoning, in which one can represent, in useful and natural ways, the knowledge in all scientific and technical disciplines. We can restate this by saying that we need formal systems in which we can express all the statements which might be made in the fields of mathematics, computer science, physics, chemistry, biology, engineering, medicine, and other physical sciences and applications of physical science. In particular, all information which is used in these fields must be representable in these logical systems.

As far as formalization is concerned, thus far mathematics has received much more attention than other disciplines. Type Theory [1] and axiomatic set theory [25] have been studied extensively as general purpose languages for expressing mathematics. If one has an information system which is based on first-order logic, one can simply regard it as being based on type theory, which includes first-order logic. If one uses some extension of axiomatic set theory, it must be an extension of a formulation such as that in [25] which accommodates entities which are not sets.

Mathematics, which is often referred to as the language of science, plays a role in all technical disciplines, so a language which is used to formalize a technical discipline should at least be adequate for formalizing mathematics. Still richer languages than those needed for mathematics may be needed to formalize certain disciplines in suitably natural ways. For example, it may be desirable to incorporate into the formal language of science some representation of the diagrams of molecular structure which chemists use. Actually, diagrams are used in mathematical reasoning too[1], and the relevant theory for Venn diagrams is well developed [13][20]. However, at present, facilities for using diagrams are not integrated into most general-purpose automated theorem proving systems.

If a variety of formal languages are developed for formalizing various disciplines, it will be natural to investigate whether some or all of them may be regarded as specializations of a single more general formal language.

Enormous progress could be made in developing automated information systems if suitable portions of natural language could be translated automatically into symbolic logic. For example, if one could translate all the theorems and definitions in Bourbaki's *Elements of mathematics* [7] into machine-readable formulas of symbolic logic, one would have a very impressive mathematical library for automated reasoning systems to use.

6 Formalization

Intellectual disciplines, such as scientific or philosophical theories, usually evolve over many years, often many centuries. Starting with isolated statements expressing various facts, they gradually become more organized and systematic as people struggle to find order in the phenomena under consideration. Gradually, concepts are formulated and general principles emerge which enable one to understand many particular facts as manifestations of these general principles. If the process progresses sufficiently far, one may obtain a systematic presentation of these general principles in which all of them can be derived from a few very basic and well specified principles called the *postulates* of the theory that has developed. At this point, one has what logicians call an *informal axiomatic theory.* If one goes a little farther, one specifies not only the postulates, but also the exact language in which the theory is to be expressed and the rules of logical reasoning which are to be used. One thus obtains what logicians call a *formal theory.* It can be seen that formalization of a theory is the ultimate act of systematization in a long chain of developments. It removes some of the last potential sources of confusion and vagueness. When one formalizes a theory, one may clarify what assumptions are tacitly made by those who habitually use the theory.

The Axiom of Choice is a striking example of an assumption which was used tacitly in mathematics for a number of years before it was explicitly recognized. The main reason that Zermelo formulated his famous list of axioms for set theory was to clarify his use of the Axiom of Choice in his proof of the Well-Ordering Theorem. The Axiom of Choice is a statement about sets, but mathematicians at that time were struggling with ambiguities in the concept of a set. [2]

[1] See [5] for persuasive arguments about using diagrams in mathematical proofs in ways that involve no compromise with complete rigor.

[2] See [14] for a fascinating and detailed discussion of this.

We need experts in various fields to formalize parts of their disciplines so that the formal theories they develop can be made part of the Universal Automated Information System, and their disciplines can benefit fully from the facilities in the system. Of course, formalization of a discipline also makes a direct contribution to the discipline by stimulating clarification and organization of the discipline.

Formalization of the knowledge in any intellectual discipline is an enormous task. However, handling the details involved in such a task and testing various aspects of the formalization are greatly facilitated by using a general-purpose theorem-proving system which can be used in a mixture of automatic and interactive modes and has suitable library facilities.

One system which satisfies these conditions is the TPS automated Theorem Proving System [2, 3, 26] which my students and I have worked on for many years. TPS also has the merit of using type theory, which in a practical sense is much more expressive than first-order logic. At conferences we generally focused on work that had been done to enhance the ability of TPS to prove theorems automatically, so some people got the impression that TPS was designed purely for automatic operation. Actually, TPS can be used automatically, interactively, or in a combination of these modes. Of course, when one is proving a nontrivial theorem, one often has to lay out the general plan of the proof interactively, but it's nice to be able to let the theorem proving system fill in the details automatically.

Automated deduction makes formalization much more tractable than it has ever been, and formalization will continue to become more tractable as theorem-proving systems improve.

7 Storing and retrieving information

When one is designing a library of information, the basic objective is to design it so that one can readily find desired information which is in the library, or determine that it is not there. It is hard to find information if one does not know how it is represented, or where it is stored in the library. Sometimes one does not really know what one is looking for, but is simply seeking information which is relevant to the problem under consideration. For example, proving a mathematical theorem is often far easier if one can find relevant theorems or lemmas to use.

To make our discussion less abstract, let's focus on the problem of designing a mathematical library in which theorems and definitions are stored as wffs of a logical language such as type theory or axiomatic set theory supplemented by facilities for using definitions.

The problem of designing good mathematical libraries is very challenging. If someone proves a mathematical theorem, and wants to find out whether it was proved previously, it often happens that the most fruitful procedure is not to use library resources, but to consult experts. It would be nice to have an objective procedure which could be automated and does not rely on human memories, judgment, or the ability to find experts who have relevant knowledge.

I'd like to discuss two basic methodologies for retrieving knowledge from a library of information. I'll refer to them as "using exhaustive search" and "using a classification system".

If we know exactly how a theorem is expressed in the library, we should be able to readily find it, and whatever information is stored with it, by exhaustive search. However, often one does not know exactly how a theorem is expressed in such a library. There are many ways of expressing mathematical theorems. For example, the statements "Every function which has an inverse is bijective" and the contrapositive form "A function which is not bijective cannot have an inverse" are clearly two ways of expressing the same theorem. Of course, some mathematicians use the terminology "one-to-one and onto" instead of "bijective".

Given wffs **A** and **B**, let's define a wff $[\mathbf{A} \equiv \mathbf{B}]$, which says that **A** and **B** are *very equivalent*, and means that **A** and **B** express the same theorem. Then if we want to find whether the theorem expressed by wff **A** is in the library, we can do an exhaustive search, checking each wff **B** in the library to see whether $[\mathbf{A} \equiv \mathbf{B}]$.

There are various ways one might define "very equivalent". We can define very weak systems of first-order logic or type theory in various ways, such as by weakening certain axioms or rules of inference. Having chosen one such very weak logical system W, we can say that **A** is very equivalent to **B** in sense W if in W we can prove that **A** is equivalent to B. Alternatively, let W be an triple consisting of some theorem-proving system, a mode of operation for that system, and a short time limit. Now we say that **A** is very equivalent to **B** in sense W if that theorem-proving system working in that mode of operation can prove that **A** is equivalent to **B** within the given time limit. [3]

Of course, normal forms are sometimes very useful. If we can define an appropriate normal form $\eta(\mathrm{A})$ for each wff A, we can say that **A** is very equivalent to **B** if and only if **A** and **B** have the same normal form. We often use the phrase normal form to mean that **A** is logically equivalent to its normal form, but one might also associate with each wff **A** a token $\tau(\mathrm{A})$ which need not be a wff logically equivalent to A, and say that **A** is very equivalent to **B** if and only if **A** and **B** have the same token. For example, the token of A might be the Skolemized form[4] of **A**. Then **A** will be very equivalent to B if **B** is obtained from **A** by moving quantifiers around in trivial ways. [5]

Of course, exhaustive search becomes increasingly unsatisfactory as libraries get very large. It is basically a very primitive procedure, even if some aspects of it are handled in very sophisticated ways. It is far better to place items in a

[3] Facilities for doing this sort of exhaustive search have been implemented for the library of the TPS automated theorem proving system by Rémy Chrétien, but have not yet been tested and released.

[4] This is to be distinguished from the *Skolem normal form* of a wff; see [8, pp. 224-227].

[5] We assume here that we are using the Skolemized form called ★ in [1, §33].

library in such a way that one will *know* where to look for them. This brings us to the topic of classification systems. If information in a library is located in a manner determined by some classification system, one can usually retrieve that information readily if one knows how it was classified. Of course, mathematical classification is done all the time. Mathematics books are classified before they are put on shelves in libraries, and articles as well as books are classified by Mathematical Reviews and Zentralblatt für Mathematik when they are reviewed. However, the classification is done by humans who use their knowledge and experience. It is not clear whether the process is sufficiently well understood that it could be mechanized.

One would like to have a classification scheme for mathematics which is based on some fundamental understanding of the structure of mathematics. It should be robust; different ways of expressing a theorem should not lead to different classifications for it. It should also be objective; we should not need to convene a committee of mathematicians to decide how a new theorem should be classified. Unfortunately, we don't yet seem to have any such fundamental understanding of the structure of mathematics.

In Hilbert's lectures from Winter term 1919, we find a very interesting passage which Wilfried Sieg translated [21, page 174] as follows:

> "The different existing mathematical disciplines are consequently necessary parts in the construction of a systematic development of thought; this development begins with simple, natural questions and proceeds on a path that is essentially traced out by compelling internal reasons. There is no question of arbitrariness. Mathematics is not like a game that determines the tasks by arbitrarily invented rules, but rather a conceptual system of internal necessity that can only be thus and not otherwise."

Hilbert was a great mathematician who understood essentially all the mathematics of his time, but he wasn't infallible, as we all know from the history of Hilbert's program. Nevertheless, maybe it will eventually become clear that his intuition on this matter was essentially correct. Perhaps there is a dimension of mathematical knowledge of which we are barely conscious: a basic structure of mathematics which can indeed be used as the rational basis for classification of mathematical theorems.

Fundamental conceptual advances are crucial steps in the growth of our understanding. We celebrate the work of Turing, Church, and others in clarifying the notion of computability. Perhaps an equally important insight into the structure of mathematical knowledge is waiting to be discovered. We really have very little basis for arguing for or against such a possibility. Perhaps it is far too early in the history of mathematics for us to have the perspective which might lead us to discover such a dimension of mathematical knowledge. Perhaps by the end of this millennium some basic facts about the structure of mathematical knowledge will have been discovered, and people will say, "All the ingredients of these ideas were known by the beginning of this millennium, but people just didn't have the perspective which was necessary for recognizing them."

For the present, it is certainly worthwhile to work on developing classification schemes for mathematics. As one example, let me mention a classification scheme for the library of the TPS automated theorem proving system which Chad Brown developed. It classifies library theorems and definitions according to the definitions which occur recursively in them. Of course, many mathematical definitions depend on other definitions. In TPS there are facilities for classifying library items automatically according to this classification scheme. While this scheme is not as robust as one might wish, it is often quite useful.

Different classification schemes may be most appropriate for different purposes, or may simply be preferred by different people. It may be useful to have a variety of classification schemes available simultaneously, and permit users to decide which classification to use at any particular time. Chad Brown implemented such a multiple classification system for the TPS library. It is described briefly in [3] and in [2]. Of course, ingenuity may be required to make multiple classification schemes work efficiently for very large libraries.

8 Planning for diversity of ideas

It is desirable to reach as much consensus as possible on basic organizational principles and languages underlying the Universal Automated Information System, so that the many components of the system can be used together in a harmonious and efficient way.

However, it is not easy to make good plans about complex projects, and people will disagree about many details. It is important to explicitly plan for ways of handling disagreements. For example, different choices of formal languages may be accommodated by suitable interfaces and translation mechanisms. In the long run, the system will evolve in ways that no one can foresee or control.

9 Reliability

The information in the Automated Information System will be contributed by many people and processes at many different times, and will involve the constantly evolving fund of human knowledge. Therefore, some of the information will be unreliable or obsolete. Systems which provide some degree of authentication for certain components of the entire information system will be needed, and automated reasoning systems will need to keep track of the sources of information they use. It may be a very complex task to develop mechanisms for safeguarding, verifying, monitoring, and assessing the reliability of information obtained from the Automated Information System.

10 Automated Reasoning Systems

Proving theorems and deriving consequences of given information are important aspects of reasoning, but reasoning involves much more. To answer a question or solve a problem, one must *find* the answer as well as prove that it is correct. This may involve search, logical deduction, and a variety of special techniques. Waldinger's paper [27] provides a nice summary of some work on deductive question-answering. One way of testing problem-solving techniques is to formalize the information in relevant textbooks, and see if the problem-solving techniques are adequate for automatically doing the exercises in those textbooks.

We anticipate that as improvements continue in systems designed to prove theorems, answer questions, and solve problems, general-purpose Automated Reasoning Systems will be developed. These will have the ability to access relevant knowledge in the Universal Automated Information System and apply logical inferences to derive answers to a great variety of questions. They will make calculations and appropriately apply algorithms, decision procedures, and special techniques for special problems. Of course, developing such systems requires a great deal of effort. The OMEGA system [23, 22, 6] is a notable example of a system which combines components with a variety of capabilities.

Automated Reasoning Systems should be able to work automatically, semi-automatically, or interactively.

11 Problems and tasks

Now let's think about some of the problems which will confront this field as we try to achieve the objectives we have been discussing:

12 Attracting students

We need to find ways to influence the education of students in scientific disciplines so that eventually workers in these fields will be familiar with formal logic and automated reasoning tools, and will be able to effectively use and contribute to those components of the Automated Information System which are relevant to their special interests.

There are many exciting potential applications of automated reasoning in addition to those we have been discussing. These include verifying software and hardware, checking mathematical proofs, and giving reasoning capabilities to robots. Someday robotic devices will need to operate in environments so complex that programmers won't be able to anticipate all the situations they may encounter. The robots will need to derive their decisions from general principles and information about the current situation.

Nevertheless, most people, even in the academic world, have very little understanding of what logic is good for. Sometimes when I meet someone, they ask what I do, and I reply "research in logic". "That's nice", they reply, and then, if they are bold enough to risk creating an embarrassing situation, they ask, "What's that good for?". They really don't know. There's a statement in the IfColog Manifesto [19] which summarizes the situation as follows: "in the wider arena of scientific activity ... logical investigations are often still perceived as limited in scope and value."

For the most part, this doesn't matter very much, but it does have one negative practical aspect. We need to have at least a few people in every technical discipline who know something about logic and are interested in formalizing at least some part of that discipline. In the long run, it will take armies of people to build the infrastructure of a Universal Automated Information System for Science and Technology. What can be done to help create an environment in which there are enough people with the knowledge and enthusiasm to make significant contributions to the development of this system?

We need to develop a variety of ideas about how to do this. One possibility might be to develop some specialized part of the Universal Automated Information System which uses automated deduction to the point where people can actually use it. It could be something fairly trivial, but it should provide an example of a situation in which it is really better to use logic and search than to depend on a fixed algorithm. I have not been able to think of a feasible idea for a particular project of this nature, but perhaps someone else can do so.

13 The problem of support for long-term projects

Another problem confronting our field, which is actually much more serious, is that it is hard to see how such an information system can be developed and sustained indefinitely in the present institutional environment. We need institutional homes for electronic libraries and certain software projects which will enable them to grow and persist indefinitely. Funding agencies often seem most interested in the development of exciting new ideas, but some projects which may be very significant in the long run require slow plodding step-by-step development which may seem quite mundane in the short run. Developing infrastructure is important, but it is sometimes difficult to claim that it is scientific research. Even when grants are made to fund the development of infrastructure, there may be little assurance that this infrastructure will be preserved indefinitely.

Perhaps one way of approaching this problem is to put it in a larger context. While it's difficult to make predictions about the future, it does seem clear that the future will bring greater complications in many realms. The world is becoming so interconnected that slight miscalculations in one part of the world can have world-wide effects. As knowledge and technology advance, we will need to have policies to guide decisions about matters which are completely or mostly out of our control now. Imagine the dilemmas which may accompany the ability to substantially manipulate climate and weather, the dilemmas which may accompany the ability to manipulate genes, the dilemmas which may accompany the ability to decipher brain waves, the dilemmas which may accompany robots which have significant abilities to function autonomously, and the dilemmas which may accompany software which controls vital aspects of our civilization but is so complex that no one understands it. What can be done now to help prepare humanity for the frightening as well as exciting complexities of the future?

There are many answers to this question, but one of them is to promote advancements in reasoning. Clearly, the ability to reason deeply, intricately, and soundly is important in dealing with all sorts of problems. But what can be done to promote the advancement of reasoning? Many thoughtful people confronted with this question may come up with some valuable ideas about improving education, but perhaps not much more. However, people in the automated reasoning community know that there is much more that can be done. We can build computerized tools which can be used to enhance our ability to reason.

Some people may need to be convinced that the use of such tools can actually enhance reasoning in significant and useful ways, but I think we can make a good case for this. Certainly work on verification is very relevant here. Another bit of evidence is provided by experience with ETPS [4], a purely interactive version of TPS which can be used by logic students to prove formal theorems. ETPS has commands for applying rules of inference and displaying and manipulating proofs, but no facilities for finding proofs automatically. One of the most difficult exercises I give to my first-semester logic students is to prove X2138:

$$\forall x \exists y\, F\, x\, y$$

$$\wedge \exists x \forall e \exists n \forall w [S\, n\, w \supset D\, w\, x\, e]$$

$$\wedge \forall e \exists d \forall a \forall b [D\, a\, b\, d \supset \forall y \forall z . F\, a\, y \wedge F\, b\, z \supset D\, y\, z\, e]$$

$$\supset \exists y \forall e \exists m \forall w . S\, m\, w \supset \forall z . F\, w\, z \supset D\, z\, y\, e$$

Typically, when this exercise is assigned to a class which must write out proofs by hand, only a few students manage to prove it correctly, but in a class using ETPS most of the students can prove it.

14 Institutes for the Advancement of Reasoning

The advancement of reasoning is a very important goal, and I would like to propose that one or more Institutes for the Advancement of Reasoning be established to promote this goal. Of course, the fact that an Institute for the Advancement of Reasoning is needed does not mean that one will be established. However, one might argue that until fairly recently the idea of an Institute for the Advancement of Reasoning did not have much substance. This is no longer true, and it seems possible that a serious proposal for the establishment of an Institute for the Advancement of Reasoning will eventually lead to the establishment of one or more such institutes. Historically, from time to time people of great wealth have established charitable institutions to support worthy causes. There are many worthy causes, and many such institutions. Perhaps some potential donor who is looking for a somewhat different but very important cause to support will decide that the advancement of reasoning is just such a cause.

What can we do to encourage the establishment of one or more Institutes for the Advancement of Reasoning? We can formulate tentative plans for such an institute which will provide a good foundation for more concrete plans. We can also articulate the value of establishing such an institute. When one or more good proposals for the establishment of an Institute for the Advancement of Reasoning have been prepared, we can circulate them appropriately and try to make it easy for potential donors to learn about them.

This is a matter which may affect many people in our community, and it would be good to have input from many people in drafting such a proposal. Jörg Siekmann and I will be working on drafting such a proposal, and we welcome suggestions about it. Please send your ideas by email to both Jörg and to me.[6] We plan to eventually set up a web site for suggestions and discussion of this matter, and link it to the IFCoLog web site http://www.ifcolog.net/. Of course, there could be a variety of such proposals; it would be fine if several people or groups prepared independent proposals.

Let's think about some of the things an Institute for the Advancement of Reasoning might do:

- Provide permanent storage for, and public access to, electronic information libraries and software related to reasoning.
- Promote and facilitate the use of automated reasoning systems.
- Have a permanent staff who will maintain automated reasoning systems, assist in their use, and conduct educational programs concerning them.
- Run summer programs for students in which they learn to use automated reasoning systems and contribute to the development of libraries of formalized knowledge.
- Provide an environment in which researchers doing work relevant to the goals of the institute can spend sabbaticals.
- Support research and projects relevant to the development and use of automated reasoning systems and the Universal Automated Information System for Science and Technology.

That's all I'll say about this topic now, but I encourage you to make suggestions about it.

15 Conclusion

Let me conclude by looking ahead from a still broader perspective. We live at the dawn of a new millennium in human history. The millennium which closed so recently featured amazing human progress, but was also replete with examples of the dire consequences of unreasonable and irrational actions. Perhaps it is not too optimistic to hope that during this new millennium mankind will reach a state where reason will generally prevail. Improvements in the ability to reason may increase the likelihood that reasoning will actually be used, and used correctly. The advancement of reasoning is an enormously important goal, and people working in the field of automated reasoning have special opportunities to help promote it.

[6] Address email to joerg.siekmann@dfki.de and to andrews@cmu.edu.

References

1. Peter B. Andrews. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof*. Kluwer Academic Publishers, second edition, 2002. Now published by Springer.
2. Peter B. Andrews and Chad E. Brown. TPS: A Hybrid Automatic-Interactive System for Developing Proofs. *Journal of Applied Logic*, 4:367–395, 2006. http://dx.doi.org/10.1016/j.jal.2005.10.002.
3. Peter B. Andrews, Chad E. Brown, Matthew Bishop, Sunil Issar, Dan Nesmith, Frank Pfenning, Hongwei Xi, Mark Kaminski, and Rémy Chrétien. TPS *User's Manual*, 2011. 142+vi pp.
4. Peter B. Andrews, Chad E. Brown, Frank Pfenning, Matthew Bishop, Sunil Issar, and Hongwei Xi. ETPS: A System to Help Students Write Formal Proofs. *Journal of Automated Reasoning*, 32:75–92, 2004. http://journals.kluweronline.com/article.asp?PIPS=5264938.
5. Jon Barwise and John Etchemendy. Visual Information and Valid Reasoning. In Walter Zimmermann and Steve Cunningham, editors, *Visualization in Teaching and Learning Mathematics*, pages 9–24. Mathematical Association of America, 1991.
6. Christoph Benzmüller, Armin Fiedler, Andreas Meier, Martin Pollet, and Jörg Siekmann. In Freek Wiedijk, editor, *The Seventeen Provers of the World*, number 3600 in LNCS, chapter OMEGA, pages 127–141. Springer, 2006.
7. Nicolas Bourbaki. *Elements of mathematics*. Hermann, 1966. 10 volumes. Translation of *Éléments de mathématique*.
8. Alonzo Church. *Introduction to Mathematical Logic*. Princeton University Press, Princeton, N.J., 1956.
9. M. Davis and H. Putnam. A Computing Procedure for Quantification Theory. *Journal of the ACM*, 7:201–215, 1960. Reprinted in [24].
10. Martin Davis. Eliminating the Irrelevant from Mechanical Proofs. In *Experimental Arithmetic, High Speed Computing and Mathematics*, Proceedings of Symposia in Applied Mathematics XV, pages 15–30. American Mathematical Society, 1963. Reprinted in [24].
11. H. Gelernter, J. R. Hansen, and D. W. Loveland. Empirical Explorations of the Geometry-Theorem Proving Machine. In *Proceedings of the Western Joint Computer Conference*, pages 143–149, 1960. Reprinted in [24].
12. P.C. Gilmore. A Proof Method for Quantification Theory. *IBM Journal of Research and Development*, 4:28–35, 1960. Reprinted in [24].
13. Eric M. Hammer. *Logic and Visual Information*. CSLI Publications & FoLLI, Stanford, California, 1995.
14. Gregory H. Moore. *Zermelo's Axiom of Choice : Its Origins, Development, and Influence*. Springer-Verlag, 1982.
15. D. Prawitz, H. Prawitz, and N. Voghera. A mechanical proof procedure and its realization in an electronic computer. *Journal of the ACM*, 7:102–128, 1960. Reprinted in [24].
16. Dag Prawitz. An improved proof procedure. *Theoria*, 26:102–139, 1960. Reprinted in [24].
17. The QED Manifesto. In Alan Bundy, editor, *Proceedings of the 12th International Conference on Automated Deduction*, volume 814 of *Lecture Notes in Artificial Intelligence*, pages 238–251, Nancy, France, 1994. Springer-Verlag.

18. J. A. Robinson. A Machine-Oriented Logic Based on the Resolution Principle. *Journal of the ACM*, 12:23–41, 1965. Reprinted in [24].
19. Dana Scott and Jörg Siekmann. IFCoLog Manifesto. http://www.ifcolog.net/?page_id=83.
20. Sun-Joo Shin. *The Logical Status of Diagrams.* Cambridge University Press, 1994.
21. Wilfried Sieg. Step by recursive step: Church's analysis of effective calculabilty. *Bulletin of Symbolic Logic*, 3:154–180, 1997.
22. Jörg Siekmann, Christoph Benzmüller, and Serge Autexier. Computer Supported Mathematics with OMEGA. *Journal of Applied Logic*, 4(4):533–559, 2006.
23. Jörg Siekmann, Christoph Benzmüller, Vladimir Brezhnev, Lassaad Cheikhrouhou, Armin Fiedler, Andreas Franke, Helmut Horacek, Michael Kohlhase, Andreas Meier, Erica Melis, Markus Moschner, Immanuel Normann, Martin Pollet, Volker Sorge, Carsten Ullrich, Claus-Peter Wirth, and Jürgen Zimmer. Proof Development with OMEGA. In Andrei Voronkov, editor, *Proceedings of the 18th International Conference on Automated Deduction (CADE-18)*, number 2392 in LNCS, pages 144–149, Copenhagen, Denmark, 2002. Springer.
24. Jörg Siekmann and Graham Wrightson, editors. *Automation of Reasoning. Vol. 1. Classical Papers on Computational Logic 1957–1966.* Springer-Verlag, 1983.
25. Patrick Suppes. *Axiomatic Set Theory.* D. Van Nostrand Company, Inc., Princeton, N.J., 1960.
26. TPS and ETPS Homepage. http://gtps.math.cmu.edu/tps.html.
27. Richard Waldinger. Whatever Happened To Deductive Question Answering? In Nachum Dershowitz and Andrei Voronkov, editors, *LPAR 2007. Proceedings Of The 14th International Conference On Logic For Programming, Artificial Intelligence And Reasoning.*, volume 4790 of *Lecture Notes in Computer Science*, pages 15–16. Springer, 2007.
28. Hao Wang. Proving theorems by pattern recognition, Part I. *Communications of the Association for Computing Machinery*, 3:220–234, 1960. Reprinted as Bell Technical Monograph 3745 and in [24].
29. Hao Wang. Toward Mechanical Mathematics. *IBM Journal of Research and Development*, 4:2–22, 1960. Reprinted in [24].
30. Hao Wang. Proving theorems by pattern recognition, Part II. *Bell System Technical Journal*, 40:1–41, 1961. Reprinted as Bell Technical Monograph 3745.