Association for Automated Reasoning

NEWSLETTER

No. 28 January 1995

From the AAR President, Larry Wos...

In keeping with our objective of having the *AAR Newsletter* serve as a forum for discussion of innovative techniques as well as controversial issues related to automated reasoning, we feature here three articles.

- The first, by D. Plaisted, proposes major changes to the current organization of CADE. Readers are strongly encouraged to respond by e-mail to Plaisted about this topic.
- The second, by Ranan Banerji, presents a new algorithm that efficiently tests theories for membership in certain decidable classes. We welcome such articles and hope that readers will find the *AAR Newsletter* a useful mechanism for publishing such results.
- The third, by William McCune, offers four challenge problems in equational logic. Faster proofs, and especially a solution to Problem D, should be sent to McCune.

The Organization of CADE

David A. Plaisted

I would like to request the help of the readers of the *AAR Newsletter* in the matter of changing the way CADE, the conference on automated deduction, is organized. Currently, all power in CADE rests in the hands of the CADE trustees, who consist of the past four program committee chairs, future already-chosen program chairs, others whom the trustees may choose to add to this committee, and the secretary and treasurer, currently 9 persons in all. Also, the trustees choose CADE program committee chairs. Therefore, as you can see, there is no democratic element at all in the current system. The program committee, the CADE conference attendees, and the members of the Association for Automated Reasoning have no voice at all in the management of CADE (except as the trustees may choose to consider it). It is true that the current trustees have shown themselves receptive to the opinions of the program committee, but there is no guarantee that this will continue. There is also the possibility that the CADE system could be adopted by other organizations (such as RTA). I propose that a more democratic system be adopted, similar to that of the Association for Logic Programming, and that an election by the members of the Association for Automated Reasoning and recent CADE attendees be held to determine whether the current system should be continued or whether a democratic system should be adopted.

I am not sure that people understand the implications of the current system, which was only recently instituted (within the past year). The trustees have essentially become a separate organization within CADE, formally insulated from the opinions of the membership at large. The trustees have taken on a life of their own. Members cannot vote for trustees or program committee chairs, and members have no say in who is chosen for program committees, except as the trustees choose to listen. The trustees could theoretically choose to do anything with CADE that they wish, and the membership at large of CADE, or even the program committee, would have no formal recourse within the current system.

I would be interested in receiving e-mail from you with your opinions about a change to a democratic system for CADE or RTA.

Proposition

I propose that all CADE trustees be elected by the members of the Association for Automated Reasoning (AAR) and by those who have attended one of the recent CADE conferences. Furthermore, that we institute a system similar to that of the Association for Logic Programming, in which members of the executive committee are elected by all members of the Association for Logic Programming, and it is the executive committee who is responsible for all financial matters and for organizing two major conferences. In their system, no one can be on the executive committee for more than two terms (4 years).

I am not proposing that program committees choose CADE program chairs; the trustees can choose. Instead, I propose that the trustees be elected.

I expect that in a democratic system, many of the same trustees would be elected as now serve. And we all recognize their contributions to the field and their valuable service to past CADE conferences.

I would also favor permitting the current CADE trustees to serve out their terms, and propose that vacancies created at the end of these terms be filled democratically. This would ensure a smooth transition to the new system.

The details of the proposed system are as follows: Elections and nominations will be held by electronic mail. The constituency for elections and nominations will be members of the AAR with e-mail access, and those who have attended one of the recent CADE conferences and have e-mail access. Nominations can be made by any member of the constituency. Each constituent can make at most one nomination. Only persons who have given consent can be nominated. All nominees with at least 10 nominations will become candidates. If there are fewer than 10 nominees with at least 10 nominations, then from the remaining nominees, those with the most nominations can be selected so that there will be 10 candidates. An election will be held among the 10 (or more) candidates, and those four receiving the largest number of votes will become trustees, serving four year terms. Each constituent may vote for up to four candidates. If the top four candidates receive a total of less than 50 percent of the vote, the next highest vote getter will become a fifth trustee. Only candidates receiving at least half of the average number of votes may become trustees, however. No position papers, distributions of biographies, or anything similar will be necessary. Elections will be held every other year, meaning that there will be a total of 8 to 10 trustees at any time. During the transitional period a slightly different schedule may be necessary. The trustees can choose among themselves such officers as may be necessary, and these officers need not be trustees. Trustees will be in charge of organizing CADE conferences, choosing CADE sites and program chairs, and other matters dealing with CADE.

I propose that the general question of adopting a democratic system, and also the question whether to adopt this specific system, be voted upon by the membership of CADE/AAR.

Discussion of the Proposition

The current system of government is unacceptable in a professional organization like CADE. It has no democratic element at all.

One may say that this issue is not important. But the lack of democratic principles can change CADE in subtle ways. Even though the current system effectively limits the terms of future trustees to about six years, with CADE meeting every year, it gives the trustees as a group too much power and too little accountability. There are many different viewpoints and subareas within CADE. What guarantee is there that they will all be treated fairly? At least a democratic system has a greater likelihood of doing this. Even if the trustees have done well so far, there is no assurance about the future. It's best to have institutionalized procedures ensuring representation by a larger CADE constituency.

The importance of this issue is also shown in that the trustees do more than choose program committee chairs. They have some authority in all matters related to the CADE conference and indirectly in all matters related to theorem proving. What if ATP really takes off and provers become very powerful? Then the trustees will assume even more importance.

Furthermore, our choice of a governmental structure can have an influence beyond CADE. It can set a precedent for other conferences as well. Thus the importance is larger than just its immediate effects on us.

Presidents of companies are responsible to stockholders, and chief editors of journals are responsible to publishing companies. But there are no clear lines of responsibility established for the trustees. The CADE by-laws do seem to state that registered attendees at CADE conferences can vote at CADE business meetings to amend the by-laws of CADE. Also, the by-laws seem to state that such business meetings should be held at each CADE conference. If such meetings are held, and if there is a real opportunity to make motions and have them voted on, then this at least provides a formal mechanism for changing the by-laws.

There is nothing within the current by-laws to prevent a subarea or subphilosophy of CADE from becoming disproportionately represented among the trustees, and thereby propagating itself indefinitely. There is no way that this could be remedied under the present system. We realize that if there were strong social pressures, the trustees would probably yield, but it is better to have an agreed-upon procedure for this. The current situation increases the temptation for the CADE trustees to act in their own self-interest rather than in the interest of the field as a whole.

This issue concerns real power, which influences the careers of every one of us working in the area of theorem proving, as well as the development of the field. This is too important an issue to let pass by. This is too much power to let rest in the hands of a nonelected body.

The trustees have said that they will consider program committee suggestions for program committee chairs very seriously. This does give a measure of representative government, at least in the choice of program committee chairs. It also sets a good precedent. However, it is dependent on the good will of the trustees and is hard to formalize or carry out, even by a well-intentioned group of trustees. It does not give the accurate representation or weight of a full program committee vote, either. Further, it does not give any representation beyond the program committee.

Do we believe that only past CADE chairs can decide who will be a good program chair? If so, then trustees, however they are elected, can consult past chairs and take their opinions into account. But in reality there are many factors to consider in choosing trustees, and being a past program chair should only be one of them that is considered. Furthermore, just because one has been a program chair, it does not follow that one has been a good program chair. All of the past program committee chairs were chosen before the trustees came into existence, and these early chairs seem to have functioned very well.

The concept of CADE as a community of friends seems to have been replaced by the concept of an organization directed by a small group of people at the top. Some organizations have very large organizing committees. Even when nondemocratic, such a large organizing committee guarantees a measure of representation by virtue of its size. The fact that CADE program chairs are chosen by a nonrepresentative group of 8 or 9 people deprives this choice of much validity.

A recent computational geometry election was held electronically with a minimum of fuss. There is no reason why CADE could not do the same.

A desire has been expressed to keep CADE informal. However, with the setting up of the CADE trustees, CADE has lost its informal nature, as those who read the legal language of the CADE by-laws can verify for themselves. We should either eliminate the trustees and go back to the previous informal structure, or establish a democratic system. Another concern has been expressed about the complexity of holding elections. However, nominations and elections can be held electronically with a minimum of trouble, as witnessed by the recent computational geometry election.

Conclusion

Thank you for considering this matter. We would like to receive e-mail from you with your views about a democratic form of government for CADE and/or RTA. Respond to plaisted@cs.unc.edu or to one of the endorsers listed below. This call for a change to a democratic form of government in CADE has been endorsed by Maria Paola Bonacina, Nachum Dershowitz, Jieh Hsiang, Gerard Huet, Michael Kohlhase, Jean-Pierre Jouannaud, Pierre Lescanne, Christopher Lynch, Dale Miller, Xumin Nie, Tobias Nipkow, David Plaisted, Wayne Snyder, Andrei Voronkov, and Hantao Zhang. In addition, six other individuals have indicated their support either for the above proposition or for the general idea of a change to a democratic system in CADE.

A Test for a Decidable Class of Theories

Ranan B. Banerji
Center for Machine Learning
Department of Mathematics and Computer Science
Saint Joseph's University
5600 City Avenue
Philadelphia, Pennsylvania 19131
(rbanerji@sju.edu)

1 Introduction

Chang and Lee [2] have discussed the completeness of Slagle's [3] semantic resolution strategy and have shown how the completeness of hyperresolution [4] (with factoring; see Leitsch [5]) as a refutation procedure, as well as the completeness of the set of support strategy [6], follows from it. Of course, semantic resolution presupposes an interpretation on the universally quantified literals. In hyperresolution one interprets the negated atoms to be true. In the set of support strategy one uses the interpretation that satisfies the satisfiable set of clauses. It is interesting to consider the fact, however, that more general classes of interpretation may be useful in increasing the efficiency of refutation. In some cases one can choose these interpretations in such a way that unsatisfiability becomes decidable. Leitsch et al. [1] have isolated a class of theories where such interpretations can be found, thus proving the decidability of certain classes of theories. The purpose of the present paper is to design an efficient algorithm for testing theories for membership in one of these classes.

In what follows, we shall introduce the necessary terminology and facts to formalize what has gone above. Our discussion presupposes knowledge of the resolution technique for refutation and its relation to the Herbrand interpretations, but is otherwise self-contained.

2 Semantic Resolution and the Decidability of PVD

Definition 2.1 *Given a set S of clauses, a Herbrand interpretation is called a* setting *if for each predicate letter P in the theory either all its Herbrand instances are true or all of them are false.*

Thus one can think of a setting as an interpretation over all the universally quantified literals. One can also think of it as dividing the predicate letters into two disjoint classes of "true" or "false" letters. So one can say the following.

Definition 2.2 *Given a setting* m*, one calls an n-place predicate letter P true iff the universal closure of $P(x_1, x_2 \ldots x_n)$ is true and false iff the universal closure of $P(x_1, x_2 \ldots x_n)$ is false under* m*.*

These definitions follow Ref. [1]. What follows are a slightly weaker form of a definition in Section 3.6 in [2] and a slightly weaker and modified form of Theorem 6.1 therein.

Definition 2.3 *Let* m *be a setting. A finite set of clauses* $\{E_1, E_2 \ldots E_q, N\}$ *with* $q \geq 1$ *is called a* semantic clash *if and only if* $E_1 \ldots E_q$ *(called* electrons *and* N *(called the* nucleus *satisfy the following conditions:*

1. $E_1 \ldots E_q$ *are false in* m.
2. *If we call* N *as* R_1 *and for each* $i = 1 \ldots q$ *there is a resolvent* $R_{(i+1)}$ *of* R_i *and* E_i *such that* $R_{(q+1)}$ *is false.*

$R_{(q+1)}$ *is called the* m-resolvent *of this clash. A sequence of clauses* $S_1, S_2 \ldots S_n$ *is called a* m-deduction *from a theory* S *if each* S_i *is either an axiom or a* m*-resolvent of previous clauses in the sequence.*

Theorem 2.1 *If* m *is a setting on an unsatisfiable theory* S*, then there is an* m-deduction *of the empty clause from* S.

The importance of settings lies in that for some m an m-deduction of the empty clause may be much shorter than for other m's. For some classes of theory m can be so chosen that all m-deductions are finite so that the theory becomes decidable. In what follows we describe such a class. The definition is according to [1], with minor rewordings to suit the present purpose.

Definition 2.4 *Given a set of clauses* S *a setting* m *is called* adequate *for* S *iff*

1. *All false clauses in* S *are ground.*
2. *If a clause* C *is true, then each variable* x *occurring in a false predicate in* C *also occurs in a true predicate in* C*, and the maximum depth of occurrence of* x *among all false predicates in* C *does not exceed the maximum depth of occurrence of* x *among all the true predicates in* C

Definition 2.5 *A set of clauses* S *a belongs to the class PVD if there exists a adequate setting for it.*

Theorem 2.2 *[1] If* S *is a PVD set of clauses and* m *is adequate for* S*, then all* m*-deductions from* S *is finite, that is, PVD classes of clauses are decidable.*

For Theorem 2.2 to be applicable one has to find an adequate setting for it. Since the total number of predicate letters in a clause set is finite, the search for such a setting (and indeed testing a clause set as PVD) is indeed finite but it need not be efficient. In what follows we propose to describe an algorithm that would do such a search in polynomial time. For such a description, we need some further definitions, discussions, and theorems. These we proceed to introduce in the next section.

3 The Test Procedure

Definition 3.1 *Given a clause C, a predicate letter P and a variable v, let $max(C, P, v)$ be the maximum depth at which v occurs inside P in clause C.*

Note that there may be more than one occurrence of P in C, and so the word "maximum" is significant.

Definition 3.2 *Given a clause C and a variable v, $maxp(C, v)$ is the set of predicates P whose $max(C, P, v)$ has the largest value among all the $max(C, P, v)$ with the same C and v.*

In what follows we use the above two definitions to make an alternative definition of PVD which can be tested in time cubic in the number of clauses and linear in the number of predicates and variables.

Definition 3.3 *A setting is called* reasonable *for a set of clauses if for all C and v either at least one member of $maxp(C, v)$ is true or all predicates in C occurring in ground atoms or involving v is false.*

The following theorem establishes the equivalence of this definition and the definition of adequate in the preceding section.

Theorem 3.1 *A setting is reasonable for a clause set S iff it is adequate for S.*

Proof. If a setting does not fulfill the second condition in the definition of adequate, then the maximum depth of some variable v among the true predicates in a clause C is strictly less than its maximum depth among the false predicates. Thus $maxp(C, v)$ would not contain any true predicate. Also, since v does occur in a true predicate in C, it negates the condition for the setting being reasonable, since in a reasonable setting v can occur only in false predicates. Hence if a setting is not adequate, then it is not reasonable.

Conversely if a setting is not reasonable, then $maxp(C, v)$ does not contain any true predicate and there is a true predicate Q involving v in C. Also, if P is any false predicate in $maxp(C, v)$ then we have v occurring in a true predicate at a strictly smaller depth in the true predicate Q than in the false predicate P. Thus the setting is not adequate. •

Thus, a theory is PVD iff it is reasonable and all the false clauses are ground.

One can test a theory for being PVD by trying to construct a reasonable setting for it in which all false clauses are ground as follows. One can calculate $maxp(C, v)$ in time proportional to the total number of occurrences of atoms in the theory times the total number of variables (just go through clauses variable by variable, keeping track of the maximum depth seen so far). Then for each ground clause one tries the feasibility of the following procedure. One sets all its predicates false. Then for each clause and variable, one sees whether $maxp(C, v)$) contains all false predicates. If any does, then all predicates in C involving v (including ground atoms) are set false. Then one goes through all clauses again with the enhanced list of false predicates. This process ends if all true clauses are exhausted or if a clause becomes false. In the latter case the process is repeated with a new ground clause. If all the ground clauses are exhausted and in each case some true clause becomes false, then the theory is not PVD. This process need not be done more than $(CV)^2$ times, where C and V are the total number of clauses and variables. After each traversal of the loop, one checks to see whether any clause is false. This takes at most CP time, where P is the total number of predicates. So $PV(CV)^2$ is the most time needed. With careful data structuring, some of the searches can be carried out in log time.

4 Comments

Thus one class of decidable theories can be tested in polynomial time. However, that does not mean that this entire class of theories can be decided in polynomial time. In [1] the authors show that the PVD class is wide enough to include some theories whose decision is reducible to NP-hard problems.

Acknowledgments

The author is indebted to Prof. Alexander Leitsch for many very illuminating discussions. Internet was of great help in making this interaction possible.

References

[1] Fernmueller, C., Leitsch, A., Tammet, T., and Zamov, N., *Resolution Methods for the Decision Problem,* Lecture Notes in Artificial Intelligence 679, Springer Verlag, New York (1994).

[2] Chang, C. L., and Lee, R. C. T., *Symbolic Logic and Mechanical Theorem Proving,* Academic Press, New York (1973).

[3] Slagle, J. R., Automatic Theorem Proving with Renamable and Semantic Resolution, *J. ACM* **14**, pp. 687–697 (1967).

[4] Robinson, J. A., Automatic Deduction with Hyperresolution, *Internat. J. Comput. Math.*,**1**, pp. 227–234 (1965).

[5] Leitsch, A., *Deciding Horn Clauses by Hyperresolution,* Lecture Notes in Computer Science bf 440, Springer Verlag, New York (1990).

[6] Wos, L., Overbeek, R., Lusk, E., and Boyle, J., *Automated Reasoning: Introduction and Applications,* 2nd ed., McGraw-Hill, New York (1992).

Four Challenge Problems in Equational Logic

William McCune (mccune@mcs.anl.gov)

The following four problems arose in work I am doing with R. Padmanabhan of the University of Manitoba. All were a challenge for our prover OTTER (we still don't have a proof for the fourth), and I'd be interested to see how other equational provers handle them. See `ftp://info.mcs.anl.gov/pub/Otter/AAR28-challenge` for a directory of OTTER input files and proofs.

Problem A. On commutators in group theory.

$$\left\{\begin{array}{l} ex = x \\ x^{-1}x = e \\ (xy)z = x(yz) \\ [x,y] = x^{-1}y^{-1}xy \\ x^3 = e \end{array}\right\} \Rightarrow \{[[[y,z],u],v] = e\}.$$

With a specialized formulation and strategy, OTTER takes several hours to find a proof.

Problem B. On semigroups.

$$\{(xy)z = x(yz),\ xy^3 = y^3x\} \Rightarrow \{(xy)^9 = x^9y^9\}.$$

With a specialized formulation and strategy, OTTER takes several minutes to find a proof.

Problem C. Quasilattice theory (QLT) is defined by

$$\left\{\begin{array}{ll} x \wedge x = x & x \vee x = x \\ x \wedge y = y \wedge x & x \vee y = y \vee x \\ (x \wedge y) \wedge z = x \wedge (y \wedge z) & (x \vee y) \vee z = x \vee (y \vee z) \\ (x \wedge (y \vee z)) \vee (x \wedge y) = x \wedge (y \vee z) & (x \vee (y \wedge z)) \wedge (x \vee y) = x \vee (y \wedge z) \end{array}\right\}.$$

The challenge is to prove one form of modularity from another:

$$\left\{\begin{array}{l} \text{QLT} \\ (x \wedge y) \vee (z \wedge (x \vee y)) = (x \vee y) \wedge (z \vee (x \wedge y)) \end{array}\right\} \Rightarrow \{x \wedge (y \vee (x \wedge z)) = (x \wedge y) \vee (x \wedge z)\}.$$

With a straightforward strategy, OTTER takes about 10 minutes to find a proof. I have not tried to prove this theorem with an associative-commutative program.

Problem D. The quasivariety HBCK is defined by

$$\left\{\begin{array}{ll} (M3)\ \ x \cdot 1 = 1 & (M8)\ \ x \cdot x = 1 \\ (M4)\ \ 1 \cdot x = x & (M9)\ \ x \cdot (y \cdot z) = y \cdot (x \cdot z) \\ (M5)\ \ (x \cdot y) \cdot ((z \cdot x) \cdot (z \cdot y)) = 1 & (H)\ \ (x \cdot y) \cdot (x \cdot z) = (y \cdot x) \cdot (y \cdot z) \\ (M7)\ \ x \cdot y = 1,\ y \cdot x = 1\ \rightarrow\ x = y & \end{array}\right\}.$$

Consider

$$(J)\quad (((x \cdot y) \cdot y) \cdot x) \cdot x = (((y \cdot x) \cdot x) \cdot y) \cdot y.$$

THEOREM (Blok and Ferreirim). HBCK *is an equational class of algebras defined by the identities* $\{M3, M4, M5, M8, M9, H, J\}$.

The challenge is to prove $\{M3, M4, M5, M7, M8, M9, H\} \Rightarrow \{J\}$. I have not been able to get OTTER to prove this. The only known proof is model theoretic.

New World Wide Web Pages at Argonne

William McCune (mccune@mcs.anl.gov)

The WWW pages on automated reasoning at Argonne were updated and extended in December. The URL of the main page is

http://www.mcs.anl.gov/Projects/otter94/otter94.html

This points to many new pages (some still under construction) on current projects, recent accomplishments and papers, history, and other automated reasoning groups and systems. In particular, the page

http://www.mcs.anl.gov/home/mccune/ar/new_results/index.html

is a summary of open questions answered and other new results obtained with our theorem provers. Let me know if you would like us to include any links to your systems or groups.

Conferences

Workshop on Executable Temporal Logics

A Workshop on Executable Temporal Logics will take place on August 19, 20, or 21, 1995, in Montreal, Canada, as part of IJCAI-95 (IJCAI registration required for workshop attendance). The workshop will cover theoretical issues in executable temporal logics, design of executable temporal logics, relationship between execution and temporal theorem-proving, operational models and implementation techniques, programming support and environments, and relationship of executable temporal logics to (temporal) databases. Attendance will be limited to approximately thirty invited participants. Those wishing to attend are encouraged to submit either (a) an extended abstract (limited to 5000 words) describing preliminary or completed work to be presented at the workshop, or, (b) single-page descriptions of research interests and current work, demonstrating the ability of the nonpresenting participants to contribute to the discussions. Submit either a LaTeX (or postscript) file by e-mail or send five paper copies by March 1, 1995, to Michael Fisher, Department of Computing, Manchester Metropolitan University, Chester Street, Manchester M1 5GD, United Kingdom; e-mail M.Fisher@doc.mmu.ac.uk; telephone: (+44) 61-247-1488; fax: (+44) 61-247-1483.

2nd International Workshop on Termination

The 2nd International Workshop on Termination will be held on May 29–31, 1995, in La Bresse, France. Topics include well-quasi orders, Kruskal's theorem; recursive path order, lexicographic path order and multiset path order; length of derivation and orders; order hierarchies; ordinals and termination; proofs by interpretation; study of hard termination problems; design and implementation of new algorithms; integration in theorem provers and experiments; and applications of termination. People who wish to present a communication are invited to submit a one-page abstract by February 20, 1995, by e-mail to termination@loria.fr.

LOPSTR'95

The 5th International Workshop on Logic Program Synthesis and Transformation will be held on September 20–22, 1995, in Utrecht, the Netherlands. Topics of interest to our readers include program synthesis, program transformation, program specialization, and partial deduction all in the context of logic programming. Also welcome are papers pointing out the relationships of the above topics with other topics in the field of automated program development, such as automated deduction, constructive type theory, implementation techniques, inductive logic programming, meta-languages, program analysis, program specification, query optimization in deductive databases, software engineering, synthesis and transformation in the context of other programming languages. Papers describing automated systems for program development and overviews of recent work on the topics of interest are also solicited. Authors should submit five copies of an extended abstract to the Program Chair by May 25, 1995: Maurizio Proietti, IASI-CNR, Viale Manzoni, 30, I-00185 Roma, Italy, Phone: +39 6 7716426, Fax: +39 6 7716461, E-mail: proietti@iasi.rm.cnr.it. E-mail submissions (LaTeX or Postscript) will also be accepted. Papers should be 5–8 pages long (excluding references and appendixes). Proofs may be added in appendix, if needed. For further information, see http://www.iasi.rm.cnr.it/../iasi/lopstr95.html.

HOA '95

A conference on Higher-Order Algebra, Logic, and Term Rewriting (HOA '95) will take place on September 21–22, 1995, in Paderborn, Germany, immediately prior to Computer Science Logic (CSL '95). The scope of the workshop includes higher-order aspects of algebra, logic and model theory; term rewriting; specification and verification languages; computational logic and theorem proving; and system implementations and case studies. Extended abstracts (up to 4 pages) of papers to be submitted should be sent to the program committee chairman Bernhard Moeller, Institut fuer Mathematik, Universitaet Augsburg, D-86135 Augsburg, Germany. E-mail: moeller@uni-augsburg.de. Fax +49 821 598 2274.

Books of Potential Interest to Our Readers

1. *Computation and Reasoning: A Type Theory for Computer Science,* Z. Luo, Oxford Science Pubs., 1994

2. *Parallel Logic Programming,* A. Takeuchi, Wiley, 1994

3. *Complexity Theory,* K. Ambos-Spies, S. Homer, and U. Schoening, eds., Cambridge, 1994

4. *Intensional Logics for Programming,* ed. L. Farinas del Cerro and M. Penttonen, Oxford Science Pubs., 1994

5. *Artificial Intelligence in Mathematics,* ed. J. Johnson, S. McKee, and A. Vella, Oxford, 1994

6. *Integrating Rules and Connectionism for Robust Commonsense Reasoning,* Ron Sun, Wiley, 1994

7. *Evidential Foundations of Probabilistic Reasoning,* D. A. Schum, Wiley, 1994

8. *Qualified Types: Theory and Practice,* Mark P. Jones, Cambridge, 1994

9. *The Gödel Programming Language,* P. Hill and J. Lloyd, MIT Press, 1994

10. *Temporal Logic: Mathematical Foundations and Computational Aspects,* D. Gabbay, I. Hodkinson, and M. Reynolds, Oxford Scientific Pubs., 1994

11. *Case-based Reasoning,* J. Kolodner, Morgan Kaufmann, 1994

12. *Predicate Logic,* R. L. Epstein, Oxford University Press, New York, 1994

If you are interested in writing a book review for the *Journal of Automated Reasoning* of any of these items, please contact Gail W. Pieper, pieper@mcs.anl.gov.

Dues Reminder

Please stay current with your dues. Your membership expiration date is printed next to your name on the mailing label of the newsletter. Dues are $7.00 for one year, $13.00 for two years, and $18.00 for three years. Members of AAR receive a substantial discount on subscriptions to the *Journal of Automated Reasoning.*

To renew your membership to AAR, send dues (in US dollars) to Bob Veroff, Dept. of Computer Science, University of New Mexico, Albuquerque, NM 87131 U.S.A. (e-mail veroff@cs.unm.edu, phone 505-277-4231)

Checks should be made payable to the Association for Automated Reasoning.

Officers of the Association for Automated Reasoning

Position	Name	E-mail
President	Larry Wos	wos@mcs.anl.gov
Vice President	Hans Jürgen Ohlbach	ohlbach@mpi-sb.mpg.de
Secretary	Robert Veroff	veroff@cs.unm.edu
Treasurer	Lawrence Henschen	henschen@eecs.nwu.edu

Contributions to the *AAR Newsletter* should be mailed to the newsletter editor, Gail Pieper: Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois 60439; pieper@mcs.anl.gov.