

# ASSOCIATION FOR AUTOMATED REASONING NEWSLETTER

No. 12

May 1989

---

From the AAR President, Larry Wos...

Our second newsletter for 1989 reflects the close link that exists in automated reasoning between theory, implementation, and application.

With regard to *theory*, Bill McCune presents a paramodulation restriction strategy for Skolem functions.

With regard to *implementation*, we announce the availability of Otter 1.0, a resolution-style theorem prover for first-order logic, and HIPER, a high-performance completion system.

With regard to *application*, we note the numerous conferences offered for the remainder of 1989—and, in particular, a call for papers for CADE-10 in 1990—that stress applications.

We continue to encourage you to send us a description of your work, as Zane Parks has done, or questions you would like answered. We note one such request here—in the article on Skolem functions—for information about a completeness proof of the restriction strategy that does not use functionally reflexive axioms.

## Conferences

### LICS'89

The fourth IEEE symposium on logic in computer science will be held on June 5-8, 1989, at Asilomar, California. Among the topics to be covered we note

- Parthenon, A Parallel Theorem Prover for Non-Horn Clauses
- Elf: A Language for Logic Definition and Verified Metaprogramming
- On the Complexity of Epistemic Reasoning
- Some Complexity Bounds for Dynamic Logic

For further information, contact Martin Abadi, DEC, Systems Research Center, 130 Lytton Ave., Palo Alto, California 94301.

### **Logic at Botic '89**

The Program Systems Institute of the U.S.S.R. Academy of Sciences is organizing a seminar on the logical foundations of computer science. The meeting will take place at Pereslavl-Zalessky July 2-9, 1989. Four invited lecturers will present talks on observational logic and process semantics and semantical programming. Other lecture topics include

- dynamic logic for program verification
- modal logics for knowledge representation systems
- synthesis of two approaches for verifying finite-state concurrent systems

For further information, write to P.O. Box 11, Pereslavl-Zalessky 152, 140 USSR.

### **Artificial Intelligence: CIVIL-COMP 89**

The interest of researchers in the use of artificial intelligence techniques in civil and structural engineering continues. The development of prototype knowledge-based systems is of particular interest. This conference, to be held in London on September 20-22, 1989, will review recent developments in artificial intelligence tools and techniques. The application topics will include

- structural analysis and design
- computer-aided design
- fault diagnosis and damage assessment
- integration with mathematical programming
- geotechnical analysis and design

For further information, contact Civil-Comp Press, 10 Saxe-Coburg Place, Edinburgh, EH3 5BR, U.K.

### **Fourth International Symposium on Methodologies for Intelligent Systems**

The Fourth Symposium is intended to attract individuals who are actively engaged both in theoretical and practical aspects of intelligent systems. The goal is to foster cross-fertilization in the following areas:

- approximate reasoning
- expert systems
- intelligent databases
- knowledge representation
- learning and adaptive systems
- logic for artificial intelligence
- neural networks

The symposium will take place on October 12-14, 1989, at Charlotte, North Carolina. For further information, please contact Dr. S.K. Michael Wong, Cornell University, Dept. of Computer Science, Upson Hall, Ithaca, NY 14853-7501.

### **Symposium on Foundations of Computer Science**

The 30th Annual IEEE Symposium on Foundations of Computer Science will be held at the Sheraton Imperial Hotel, Research Triangle, North Carolina, October 30 - November 1, 1989. Papers presenting original research on theoretical aspects of computer science are sought in the following areas:

- Algorithms and data structures
- Automata and formal languages
- Computability and complexity theory
- Computational geometry and robotics
- Cryptography
- Databases
- Logic in computer science
- Machine Learning
- VLSI computation and design
- Parallel and distributed computation

Those wishing further information should write to the program committee chair: Zvi Galil, Dept. of Computer Science, Columbia University, 450 Computer Science Bldg., 500 West 120th Street, New York, NY 10027.

### **Advances in Intelligent Robotics Systems**

“Advances in Intelligent Robotics Systems” really comprises two conferences:

1. Algorithms and techniques for intelligent robotics, and
2. Systems and applications.

Topics for both conferences include artificial intelligence, pattern recognition, and fuzzy logic in intelligent systems.

The conferences will take place on November 5-10, 1989, at the Adams Mark Hotel in Philadelphia, Pennsylvania. For further information, write to SPIE Technical Program Committee, Philadelphia'89, P.O. Box 10, Bellingham, WA 98227-0010.

## CADE-10

The Tenth International Conference on Automated Deduction will be held in West Germany on July 23-27, 1990. CADE is the major forum at which research on all aspects of automated deduction can be presented. Papers on automated deduction (for classical and nonclassical logics) in the following and related fields are invited:

Theorem Proving Unification	Decision Procedures Program Verification and Synthesis	Logic Programming Inference Systems
Term Rewriting	Deductive Databases	Applications

### *Program Committee*

Peter Andrews	Claude Kirchner	William Pase
Wolfgang Bibel	Jean-Louis Lassez	Lawrence Paulson
W.W. Bledsoe	Donald Loveland	Fernando Pereira
Alan Bundy	Ewing Lusk	David Plaisted
Robert Constable	Michael McRobbie	Joerg Siekmann
Jean-Pierre Jouannaud	Dale Miller	Mark Stickel, <i>Chairman</i>
Deepak Kapur	Hans Juergen Ohlbach	Richard Waldinger
Matt Kaufmann	Ross Overbeek	Christoph Walther

Original research papers (up to 5,000 words; 15 proceedings pages, 6 by 9 inches, 12-point type, will be allotted) are solicited. Also solicited are system summaries that describe working reasoning systems (2 proceedings pages) and problem sets that provide realistic, interesting challenges for automated reasoning systems (5 proceedings pages). The title page of the submission should include author's name, address, phone number, and E-mail address. Papers must be unpublished and not submitted for publication elsewhere. Late papers and papers that require major revision, including submissions that are too long, will be rejected.

Submission receipt deadline:	November 27, 1989
Author notification date:	February 15, 1990
Camera-ready copy receipt deadline:	April 2, 1990

Six paper copies should be sent to arrive by November 27, 1989, to

Mark E. Stickel  
Artificial Intelligence Center  
SRI International  
333 Ravenswood Avenue  
Menlo Park, CA 94025 U.S.A.

Inquiries about CADE can also be sent by electronic mail to [Stickel@AI.SRI.COM](mailto:Stickel@AI.SRI.COM)

Looking for Single Axioms for  $E_{\rightarrow}$ ,  $R_{\rightarrow}$ , and  $RM_{\rightarrow}$   
(Zane Parks)

I am investigating the problem of finding single-axiom formulations of  $E_{\rightarrow}$ ,  $R_{\rightarrow}$ , and  $RM_{\rightarrow}$ . This overlaps a problem set in [1, p. 89]. I am using a personal computer, but no automated reasoning yet. Still, my approach may be of interest to readers of this newsletter.

The sequence  $E_{\rightarrow}$ ,  $R_{\rightarrow}$ ,  $RM_{\rightarrow}$  was chosen because each is contained in the next and  $RM_{\rightarrow}$  is three valued. General properties of propositional systems [1, pp. 83-84], together with variable-sharing properties of these systems, imply that a single axiom be at least thirteen letters long and contain three variables, say,  $p$ ,  $q$ , and  $r$ , with three occurrences of  $p$  and two each of  $q$  and  $r$ .

All such formulae can be generated by a program and tested for validity on  $RM_{\rightarrow}$ 's three-valued matrix. The resulting formulae are in turn tested for invalidity on a trio of matrices falsifying Contraction  $CCpCpqCpq$ , Identity  $Cpp$ , and Transitivity (Suffixing)  $CCpqCCqrCpr$ , respectively. (Only invalid formulae could have these common theorems of  $E_{\rightarrow}$ ,  $R_{\rightarrow}$ , and  $RM_{\rightarrow}$  as consequences.)

No thirteen-letter formula passes all these tests. So, a single axiom must be at least fifteen letters long. A fifteen-letter formula would have to contain three variables, say,  $p$ ,  $q$ , and  $r$  with three occurrences of  $p$  and  $q$  and two of  $r$ . Again, these can be generated and tested by programs. In this case, more extensive testing is required, but the end result is the same—no fifteen-letter formula passes all tests.

Details (including programs) may be found in [2]. I am currently investigating the case of seventeen-letter formulae.

## References

- [1] Anderson, A. R., and Belnap, N. D., *Entailment: The Logic of Relevance and Necessity*, Vol. 1, Princeton University Press, Princeton (1975).
- [2] Parks, Z., "The Quest for Single Axiom Formulations of  $E_{\rightarrow}$ ,  $R_{\rightarrow}$ , and  $RM_{\rightarrow}$ ," forthcoming.

### Skolem Functions and Equality

(William McCune, Argonne National Laboratory, [mccune@mcs.anl.gov](mailto:mccune@mcs.anl.gov))

With respect to equality, Skolem functions should be handled differently from ordinary functions. Some researchers in automated deduction know this fact, but I have not seen much written on the topic. This note contains some observations which lead to a paramodulation restriction strategy which I have not seen before. The strategy prohibits paramodulation into a proper subterm of a Skolem term (a term whose leading function symbol is a Skolem function symbol). There is an intuitive justification as well as an completeness justification for the strategy.

Let  $F$  be a (quantified) first-order formula. It is well known that  $F$  is E-unsatisfiable if and only if  $F \ \& \ eq\_ax(F)$  is unsatisfiable. ( $eq\_ax(F)$  is the standard set of equality axioms for  $F$ ). If  $sk(G)$  denotes a Skolem form of  $G$ , then the following four statements are equivalent:

- (1)  $F$  is E-unsatisfiable;
- (2)  $sk(F)$  is E-unsatisfiable;
- (3)  $F \ \& \ eq\_ax(F)$  is unsatisfiable;
- (4)  $sk(F \ \& \ eq\_ax(F))$  is unsatisfiable.

The point is that the equality axioms can be fixed before Skolemization occurs, with the result that equality substitution axioms are not required for Skolem functions.

Many introductions to automated deduction (for example, [1,3]), fail to mention this fact, and some (for example, [2]) cover it only in part. In addition, many of the standard clause sets that people use to evaluate their theorem provers contain equality substitution axioms for Skolem functions.

The fact that equality substitution axioms are not needed for Skolem functions can be turned into a strategy for restricting paramodulation. Let  $f$  be a 3-place function symbol, and consider a term  $f(t1, t2, t3)$ . Paramodulation into a subterm of  $t2$  corresponds to a sequence of resolution steps with equality axioms. One of the equality axioms in question is second position substitution for  $f$ , because paramodulation is into a subterm of the second argument of  $f$ . If  $f$  is a Skolem function, that equality axiom need not be present, indicating that the paramodulation inference need not be made.

The strategy is: do not paramodulate into any proper subterm of a Skolem term (a term that starts with a Skolem function). Note that condition *proper*: the strategy does not prohibit paramodulation into (substitution for) the entire term  $f(t1, t2, t3)$  ( $f$  a Skolem function.)

It is not difficult so show that the restriction strategy is complete. Let  $S$  be an E-unsatisfiable set of clauses, and let  $eq\_ax$  be the equality axioms for the symbols (excluding Skolem symbols) in  $S$ . Let  $R$  be a hyperresolution/factoring refutation of  $S$ . If one transforms  $R$  into a hyperresolution/paramodulation/factoring refutation by replacing all hyperresolution steps involving equality axioms with paramodulation steps, none of the paramodulation steps violate the restriction. However, the straightforward transformation procedure requires the use of the functionally reflexive axioms. I have tried unsuccessfully to adapt existing proofs of the completeness of paramodulation without the functionally reflexive axioms, and I would be interested in hearing from anyone who knows a completeness proof of the restriction strategy that does not use the functionally reflexive axioms.

The restriction strategy has intuitive appeal as well. The arguments of Skolem function should serve merely as place-holders for the objects on which the "existing" object depends—equality substitution should not be applied to them. As the search progresses, the arguments

become more and more instantiated and should serve as constraints on unification rather than being replaced with equal objects.

I have constructed (by hand) many set theory, topology, and combinatory logic paramodulation proofs involving Skolem functions. In *none* of the proofs did it seem necessary of natural to paramodulate into subterms of Skolem expressions. The next step is to try it with an automated theorem prover.

## References

- [1] Chang, C.-L., and Lee, R. C.-T., *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York, 1973.
- [2] Loveland, D., *Automated Theorem Proving: A Logical Basis*, North-Holland, Amsterdam, 1978.
- [3] Wos, L., Overbeek, R., Lusk, E., and Boyle, J., *Automated Reasoning: Introduction and Applications*, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

## HIPER

(Jim Christian, University of Texas)

A new E-completion system, called HIPER (for High-Performance Rewriting), is now available. HIPER is written in Austin Kyoto Common Lisp (also available on rascal), though it should run in other Common Lisps with little or no change.

HIPER has a number of interesting features, including the following:

- Discrimination tree indexing and a new linearized term representation. These yield excellent performance—typically 20 to 30 times speedup, and asymptotically more. HIPER is believed to be the fastest completion system available anywhere. Processing rates on a Sun 4 range from a hundred to a thousand equations per second on a wide range of problems.
- Dynamic modification of the unification and termination algorithms to handle failure equations. The new system handles simple linear permutative theories, instead of the theories associated with Claude Kirchner's unification tools. Simple linear permuters combined with new function symbol introduction seem to provide the same amount of power as the Kirchner theories, but the completion procedure is much simpler and faster.
- Unfailing completion. This allows rewriting and critical pairs with orientable instances of non-orientable equations.
- Heuristically-guided function symbol introduction. HIPER can find complete sets for several axiomatizations of groups due to Higman, with no user intervention.
- Theorem-proving abilities, including the set of support strategy.

To obtain HIPER, use ftp to connect to rascal.ics.utexas.edu. Use "anonymous" as the user name. Any password will suffice, though it is considered courteous to use your net address. Type "cd pub/jimc" to get to the HIPER files. The file README will tell how to transfer HIPER. Included in the HIPER directory is a user's manual and a copy of the paper that will appear in the RTA-89 proceedings.

### Availability of Otter 1.0

(William McCune, Argonne National Laboratory, [mccune@mcs.anl.gov](mailto:mccune@mcs.anl.gov))

The theorem prover Otter 1.0 has been released and is in the public domain (as with the Boyer-Moore Theorem Prover—no fee, no license, no support).

Our site does not have anonymous FTP, but Otter 1.0 (including the manual and test problems) can be obtained by FTP from

herky.cs.uiowa.edu -- username anonymous, password guest, go to public/otter and follow the directions in README.

dopey.cs.unc.edu -- username anonymous, password guest, go to pub/Otter and follow the directions in README.

Starting in the summer of '89, PC diskettes of Otter will probably be distributed also by *Books and Bytes*, (312) 416-0102. Along with an MS-DOS executable file, the PC diskettes contain essentially the same material as the normal UNIX release, so if you can get the diskette files onto your UNIX system, you will be able to compile and run Otter there as well.

Otter 1.0 is now being distributed (on tar tapes and PC diskettes) by the following organizations. (Pricing policies vary.)

Numerical Algorithms Group Inc.  
Suite 100  
1101 31st St.  
Downers Grove, IL 60515  
phone: (312) 971-2337

National Energy Software Center  
Argonne National Laboratory  
Argonne, IL 60439  
phone: (312) 972-7172

Otter is similar in scope and purpose to AURA and LMA/ITP, which are also closely related to Argonne. Here is the abstract from the Users' Guide:

Otter (Other Techniques for Theorem-proving and Effective Research) is a resolution-style theorem-proving program for first-order logic with equality. Otter includes the inference rules binary resolution, hyperresolution, UR-resolution, and binary paramodulation. Some of its other abilities are conversion from first-order formulas to clauses, forward and back subsumption, factoring, weighting, answer literals, term ordering, forward and back demodulation, and evaluable functions and predicates. Otter is coded in C, and it is portable to a wide variety of computers.