The IJCAR-2004 Automated Theorem Proving Competition

Geoff Sutcliffe
Department of Computer Science
University of Miami
USA
E-mail: geoff@cs.miami.edu

The CADE ATP System Competition (CASC) is an annual evaluation of fully automatic, first order Automated Theorem Proving systems – the world championship for such systems. CASC-J2 was the ninth competition in the CASC series. Twenty-six ATP systems and system variants competed in the various competition and demonstration divisions. An outline of the competition design, and a commentated summary of the results, are presented.

Keywords: competition, automated theorem proving

1. Introduction

The CADE ATP System Competition (CASC) is an annual evaluation of fully automatic, first order Automated Theorem Proving (ATP) systems – the world championship for such systems. CASC has the primary aim of evaluating the relative capabilities of ATP systems. Additionally, CASC aims to stimulate ATP research in general, to stimulate ATP research towards autonomous systems, to motivate implementation and fixing of systems, to provide an inspiring environment for personal interaction between ATP researchers, and to expose ATP systems both within and beyond the ATP community. CASC-J2 was held on 6th July 2004, as part of the 2nd International Joint Conference on Automated Deduction, in Cork, Ireland.[1] CASC-J2 was the ninth competition in the CASC series; see [13] and citations therein for information about the previous competitions. Twenty-six ATP systems and system variants, listed in Table 1, competed in the various competition and demonstration divisions of CASC-J2. The division winners of CASC-19 (the previous CASC) were automatically entered to provide benchmarks against which progress can be judged. Details of the CASC-J2 design, and system descriptions for the entered systems, are in [12] and on the CASC-J2 WWW site. The WWW site also provides access to all systems and competition resources:
`http://www.cs.miami.edu/~tptp/CASC/J2/`

CASC-J2 was organized by Geoff Sutcliffe, and was overseen by a panel consisting of Alan Bundy, Uli Furbach, and Jeff Pelletier. The competition machines were supplied by the University of Manchester.

This paper is organized as follows: Sections 2 and 3 describe the divisions and organization of CASC-J2. Section 4 provides a commentated summary of the results, and short descriptions of the division winners are given in Section 5. Section 6 concludes and discusses plans for future CASCs.

2. Divisions

CASC is run in divisions according to system and problem characteristics. In CASC-J2 there were five *competition divisions*, in which the systems were ranked according to the numbers of problems solved, with ties decided by average CPU times over problems solved.

- The **MIX** division used mixed CNF really non-propositional theorems (unsatisfiable clause sets). *Mixed* means Horn and non-Horn problems, with or without equality, but not unit equality problems (see the UEQ division below). *Really non-propositional* means with an infinite Herbrand universe. The MIX division had five problem categories: **HNE** - Horn with No Equality, **HEQ** - Horn with some (not

[1] In 2004 CADE was part of the 2nd International Joint Conference on Automated Reasoning, hence “J2” for “2nd Joint conference”.

AI Communications
ISSN 0921-7126, IOS Press. All rights reserved

Table 1
The ATP systems and entrants

ATP System	Divisions	Entrants	Affiliation
Darwin CASC-J2	MIX SAT* EPR (demo)	Alexander Fuchs, Peter Baumgartner Cesare Tinelli	Universität Koblenz-Landau University of Iowa
DCTP 1.3-EPR	EPR		*CASC-19 EPR winner*
DCTP 1.31	MIX	Gernot Stenz	Max-Planck-Institut für Informatik
DCTP 1.31-SAT	SAT		*DCTP 1.31 variant*
DCTP 1.31-EPR	EPR		*DCTP 1.31 variant*
DCTP 10.21p	MIX FOF EPR	Gernot Stenz	Max-Planck-Institut für Informatik
DCTP 10.21p-SAT	SAT		*DCTP 10.21p variant*
Dilemma 0.1	FOF	Magnus Björk	Chalmers University of Technology
E 0.82	MIX FOF UEQ	Stephan Schulz	Technische Universität München & IRC/irst
EP 0.82	MIX* FOF*		*E 0.82 variant*
E-SETHEO csp04	MIX FOF EPR UEQ	Gernot Stenz, Reinhold Letz, Stephan Schulz	Max-Planck-Institut für Informatik Technische Universität München
E-S'O csp04-SAT	SAT		*E-SETHEO csp04 variant*
Gandalf c-2.6-SAT	SAT		*CASC-19 SAT winner*
Mace2 2.2	SAT*	William McCune	Argonne National Laboratory
Mace4 2002-D	SAT*	William McCune	Argonne National Laboratory
Octopus 2004	MIX (demo)	Monty Newborn, Zongyan Wang	McGill University
Otter 3.3	MIX* FOF UEQ	William McCune	Argonne National Laboratory
Paradox 1.0	SAT*		*CASC-19 SAT winner*
Paradox 1.1-casc	SAT* EPR	Koen Claessen, Niklas Sörensson	Chalmers University of Technology
SOS 1.0	MIX* UEQ	John Slaney, Arnold Binas, David Price	Australian National University
THEO J2004	MIX*	Monty Newborn	McGill University
Vampire 5.0	FOF		*CASC-19 FOF winner*
Vampire 6.0	MIX*		*CASC-19 MIX winner*
Vampire 7.0	MIX* FOF* EPR UEQ	Andrei Voronkov, Alexandre Riazanov	The University of Manchester
Waldmeister 702	UEQ		*CASC-19 UEQ winner*
Waldmeister 704	UEQ	Thomas Hillenbrand, Jean-Marie Gaillourdet, Bernd Löchner	Max-Planck-Institut für Informatik Universität Kaiserslautern

MIX* indicates participation in the MIX division proof class, FOF* indicates participation in the FOF division proof class, and SAT* indicates participation in the SAT division model class - see Section 2.

pure) Equality, **NNE** - Non-Horn with No Equality, **NEQ** - Non-Horn with some (not pure) Equality, and **PEQ** - Pure Equality.

- The **FOF** division used non-propositional first-order form theorems (axioms with a provable conjecture). The FOF division had two problem categories: **FNE** - FOF with No Equality, and **FEQ** - FOF with Equality.
- The **SAT** division used CNF really non-propositional non-theorems (satisfiable clause sets). The SAT division had two problem categories: **SNE** - SAT with No Equality, and **SEQ** - SAT with Equality.
- The **EPR** division used CNF effectively propositional theorems and non-theorems. *Effectively propositional* means syntactically non-propositional but with a finite Herbrand universe. The EPR division had two problem categories: **EPT** - Effectively Propositional Theorems (unsatisfiable clause sets), and **EPS** - Effectively Propositional non-theorems (Satisfiable clause sets).
- The **UEQ** division used unit equality CNF really non-propositional theorems (unsatisfiable clause sets).

The MIX, FOF, and SAT divisions each had two ranking classes: an **assurance** class - ranked according to the number of problems solved (a “yes” output, giving an *assurance* of the existence of a proof/model), and a **proof/model** class - ranked according to the number of problems solved with an *acceptable proof/model output*. The competition panel judged whether or not each system's proof/model format is acceptable.

Additionally, CASC has a *demonstration division*, in which systems demonstrate their abilities without being formally ranked, using the same problems as the competition divisions.

3. Organization

The CASC-J2 competition divisions were run on 60 AMD Athlon XP 2200+ computers, each having a 1.8 GHz CPU, 512 MB memory, and the Linux 2.4.9-20 operating system. In the demonstration division, Darwin ran on an Intel P4 computer, with a 2.4GHz CPU, 512 MB memory, and the Linux 2.4.21-231 operating system. Octopus ran on a network of 120 computers, each having an Intel P3 or P4 CPU, between 256 MB and 512 MB memory, and either the FreeBSD or Linux operating system.

The problems were taken from the TPTP problem library [14], v2.7.0. TPTP v2.7.0 was not released until after the competition, so that new problems had not previously been seen by the entrants. Unbiased TPTP problems with a TPTP difficulty rating in the range 0.21 to 0.99 were eligible for use. The problems used were randomly selected from the eligible problems, based on a seed provided by the panel at the start of the competition. The random selection was subject to a limitation on the number of very similar problems in each division and category [11], and biased to ensure (if possible) the selection of at least 50% new problems in each division and category. Table 2 gives the numbers of eligible problems, the maximal numbers that could be used after taking into account the limitation on very similar problems, and the numbers of problems used, in each division and category.[2] Only the FEQ and NEQ categories had significant numbers of new problems. Due to the small maximal numbers of usable problems in the EPT and EPS categories, the limitation on the number of very similar problems could not be fully imposed in the EPR division. To ensure that no system received an advantage or disadvantage due to the specific presentation of the problems in the TPTP, the `tptp2X` utility was used to replace all predicate and function symbols with new symbols, randomly reorder the formulae and the clauses' literals, and randomly reverse the unit equalities in the UEQ problems.

The ATP systems were required to be sound and fully automatic. The organizers tested the systems for soundness by submitting non-theorems to the systems participating in the MIX, UEQ, FOF, and EPR divisions, and theorems to the systems participating in the SAT and EPR divisions. Claiming to have found a proof of a non-theorem or a disproof of a theorem indicates unsoundness. Two systems failed this test, and were repaired in time for the competition. Additionally, after the competition the entrants of Darwin discovered that Darwin was unsound with respect to unsatisfiability, and Darwin was thus retrospectively disqualified from the competition by the competition panel. It should be noted that the unsoundness occurred only for certain types of problems, and that there was no intention to deceive. A repaired version of Darwin was retrospectively entered into the demonstration division, and those results are reported in this paper. Fully automatic operation meant that any command line switches had to be the same for all problems.

A 600s CPU time limit was imposed on each solution attempt. In the demonstration division, Darwin used a 500s so that its results on the slightly faster computer, are reasonably comparable with the competition division results. The Octopus entrant decided to reduce its time limit to 400s, in order to complete the competition in a timely fashion. A wall clock time limit of double the CPU time limit was imposed in all the competition divisions, to limit very high memory usage that causes swapping.

4. Results

For each ATP system, for each problem, three items of data were recorded: whether or not the

[2] It was originally intended that there would be 35 FNE problems. However a bug in the problem classification software was discovered after the event, resulting in 12 problems with equality in the FNE category. Those problems have been removed from the competition results.

Table 2
Numbers of eligible and used problems

Division	MIX					FOF		SAT		EPR		UEQ
Category	HNE	HEQ	NNE	NEQ	PEQ	FNE	FEQ	SNE	SEQ	EPT	EPS	
Eligible	174	112	119	599	28	113	568	154	136	46	94	139
Eligible new	0	1	1	19	2	33	82	3	6	1	0	1
Max usable	174	61	101	444	28	63	568	154	136	32	30	139
Max new	0	1	1	19	2	22	82	3	6	1	0	1
Used	35	35	35	75	20	23	65	50	50	40	40	100
New used	0	1	1	16	2	1	13	3	6	1	0	1

problem was solved, the CPU time taken, and whether or not a solution (proof or model) was output. This section summarizes the results, and provides some commentary. Detailed results, including the systems' output files, are available from the CASC-J2 WWW site. In each of the results summary tables below, the CASC-19 winner is *emphasized.*

4.1. The MIX Division

Tables 3 and 4 summarize the results in the MIX division. As Vampire outputs proofs, Vampire was the winner of both the Assurance and Proof classes. The improved performance of the top four systems over Vampire 6.0, the CASC-19 winner, indicates progress in the area. The improved performance of Vampire 7.0 over Vampire 6.0 is due to several improvements in the underlying reasoning mechanisms and extensive experimental evaluation leading to more aggressive use of strategy scheduling (see Section 5 for details). E-SETHEO, which internally uses several distinct ATP systems, benefited largely from the use of E, which solved 81 of the 174 problems. E and EP are the only non-strategy scheduling systems in the top five, selecting a single strategy based on problem characteristics. Note that for five problems, the postprocessing in EP could not be completed within the time limit, even though it had been determined that a proof exists.

There is a reasonably large gap between the top five systems and the lower ranked systems. This divergence in the MIX division was discussed in the report on CASC-19 [13], and is even more pronounced here. The split is particularly acute in the HEQ and PEQ categories.

Table 3
MIX division results

ATP System	MIX /200	Avg time	Prfs out	New /20
Vampire 7.0	180	51.4	180	14
E-SETHEO csp04	174	36.0	0	13
E 0.82	162	26.4	0	7
EP 0.82	161	27.7	156	7
Vampire 6.0	157	80.3	157	12
DCTP 10.21p	103	33.2	0	10
THEO J2004	83	73.3	82	11
DCTP 1.31	66	17.1	0	11
SOS 1.0	39	124.2	39	3
Otter 3.3	37	74.6	37	3
Demonstration division				
Darwin CASC-J2	44	-	0	7
Octopus 2004	115	-	-	15

The rankings in the categories align quite closely with the division ranking, with the exception of Vampire 6.0's relatively strong performance in the HNE category (an artifact of stronger performances by the other systems in the non-HNE categories). In previous CASCs the E and EP systems outperformed the other systems in the HEQ category, but this superiority is not evident in these results, where the top four systems all solved 31 of the 35 HEQ problems. Between Vampire 7.0 and E, all the HEQ problems were solved.

The ranking according to the number of new problems solved aligns well with the overall ranking, except for E and EP. This may indicate some over-tuning of E to existing TPTP problems. It is noteworthy that two lower ranked systems, DCTP and THEO, performed well on the new problems. In the demonstration division, Octopus solved 15 of the new problems, more than any of the competition division systems.

The individual problem results show that three problems were solved by all the systems, and three problems - `COL090-2`, `SYN076-1`, `SYN314-1.002.001`

Table 4
MIX category results

ATP System	HNE /35	HEQ /35	NNE /35	NEQ /75	PEQ /20
Vampire 7.0	35	31	34	64	66
E-SETHEO csp04	34	31	31	62	16
E 0.82	31	31	32	55	13
EP 0.82	30	31	32	55	13
Vampire 6.0	35	20	28	60	14
DCTP 10.21p	27	8	22	45	1
THEO J2004	19	1	19	43	1
DCTP 1.31	19	3	12	31	1
SOS 1.0	12	6	4	13	4
Otter 3.3	13	3	3	15	3
Demonstration division					
Darwin CASC-J2	15	0	14	13	2
Octopus 2004	27	4	24	58	2

- were unsolved. These three problems were eligible because they have been solved by systems that were not entered into the MIX division. Fourteen problems were solved by only Vampire (nine by both versions 6.0 and 7.0, and five by only version 7.0). The only other unique solutions in the MIX division were by E-SETHEO, DCTP, and THEO, each of which solved one NEQ problem that no other system solved. There was one problem - `LAT036-1` - that was solved only by SOS and Otter, the two lowest ranked systems in the division. In the demonstration division, Octopus solved three problems - `COL090-2`, `COL093-2`, `SYN076-1` - that were not solved by any competition division system.

4.2. The FOF Division

Table 5 summarizes the results in the FOF division. As Vampire outputs proofs, Vampire was the winner of both the Assurance and Proof classes. All the systems except Dilemma (a prototype system based on an extension of Stålmarck's method to first order classical logic) work by converting to CNF and producing a refutation.

The individual problem results show that Vampire 7.0 solved all but two of the problems that were solved by all the other systems combined, and the Vampires solved six problems that no other system solved. Six problems, all FEQ, were unsolved. They were eligible because they can be solved by SPASS [15]. Four of the six unsolved problems were new `ALG` problems, from the automated classification of finite algebraic structures [4]. These four problems are pure equality problems containing very large formulae, and none of the CNF-based systems could convert the problems to CNF.

Table 5
FOF division and category results

ATP System	FOF /88	Avg time	Prfs out	FNE /23	FEQ /65
Vampire 7.0	80	34.7	80	22	58
Vampire 5.0	75	19.3	75	22	53
E-SETHEO csp04	74	18.0	0	23	51
E 0.82	72	17.2	0	22	50
EP 0.82	72	21.2	71	22	50
DCTP 10.21p	52	30.4	0	19	33
Otter 3.3	23	34.1	23	4	19
Dilemma 0.1	11	0.1	0	9	2

As in the MIX division, there is a notable split between the top five systems and the lower ranked systems. This is not surprising, given that the systems convert to CNF and then proceed as for the MIX division.

4.3. The SAT Division

Table 6 summarizes the results in the SAT division. The CASC-19 winners, Gandalf c-2.6-SAT (Assurance class) and Paradox 1.0 (Model class) won again. However, the new Paradox 1.1-casc is even faster than the previous version, due to a reimplementation of the ground instantiation procedure in C++ (previously implemented in Haskell), using a more sophisticated instantiation algorithm. The Gandalf and Paradox systems performed notably better than the other systems, producing a split as in the MIX and FOF divisions. There were 9 new problems in the SAT division, of which Gandalf solved all, Paradox 1.0 solved 8, and Paradox 1.1-casc solved 7. None of the other systems solved more than 4 of the new problems, suggesting that the techniques used in Gandalf and Paradox extend more usefully to new unseen problems. Two problems - `LCL415-1` and `SWV021-1` - were solved by only E-SETHEO, which is noteworthy in light of it's weaker overall performance.

4.4. The EPR Division

Table 7 summarizes the results in the EPR division. In contrast to CASC-19 where the mono-

Table 6

SAT division and category results

ATP System	SAT /100	Avg time	Mdls out	SNE /50	SEQ /50
Gandalf c-2.6-SAT	95	121.9	65	48	47
Paradox 1.0	94	5.0	94	49	45
Paradox 1.1-casc	92	2.9	92	49	43
Mace4 2004-D	55	0.9	55	25	30
Mace2 2.2	46	9.0	46	16	30
E-SETHEO csp04-SAT	46	17.8	0	27	19
DCTP 10.21p-SAT	42	16.1	0	25	17
DCTP 1.31-SAT	35	0.0	0	19	16
Demonstration division					
Darwin CASC-J2	14	-	14	10	4

lithic DCTP 1.3-EPR won, the strategy scheduling version of DCTP won the division. One of the two strategies available in DCTP 10.21p (see Section 5) was particularly effective, solving 31 EPT and 25 EPS problems. Within E-SETHEO, that same DCTP strategy solved 72 of the 79 solved problems. Only Vampire could output solutions in both problem categories, while Paradox and Darwin output models in the EPS category. The strong performance of Darwin, a new implementation [1] of the new model evolution calculus [2] was noteworthy, and gained the system the "Outstanding Newcomer" award.

The individual problem results show that 19 EPT problems but no EPS problems were solved by all the systems. This indicates that some of the eligible EPT problems were too easy for comparing the systems. Paradox was the only system to solve the one new problem in the division.

Table 7

EPR division and category results

ATP System	EPR /80	Avg time	Ps/Ms output	EPT /40	EPS /40
DCTP 10.21p	79	26.5	0/0	39	40
E-SETHEO csp04	79	38.3	0/0	39	40
DCTP 1.31-EPR	72	36.1	0/0	35	37
DCTP 1.3-EPR	72	66.8	0/0	35	37
Paradox 1.1-casc	56	39.9	0/28	27	29
Vampire 7.0	46	18.0	37/9	37	9
Demonstration division					
Darwin CASC-J2	69	-	0/37	32	37

4.5. The UEQ Division

Table 8 summarizes the results in the UEQ division. The winner, Waldmeister 704, outperformed Waldmeister 702, which had won the UEQ division of the two previous CASCs. The improved performance of Waldmeister 704 is due to improved detection of redundancies in the presence of associative-commutative operators, implementation changes, and improved strategies. As noted in [13], for many years prior to CASC-19 Waldmeister had dominated the UEQ division, but this domination was not evident in CASC-19. These results indicate that the domination may have returned.

The individual problem results show that 19 problems were solved by all the systems. These problems were eligible because they had not been solved by some systems that were not entered into CASC-20, but which had solved the lattice theory problems that were new in CASC-19 [13]. Those problems were unsolved by Waldmeister 703 (the Waldmeister version entered into CASC-19), but can be solved by Waldmeister 704. Waldmeister 703, and the systems that solved the lattice theory problems, contributed to the problem ratings used in CASC-20. This meant that some problems that could be solved by all the systems were eligible for CASC-20. Twelve problems were solved by only the two Waldmeister versions (none were solved by only the latest version 704), and a further five problems were solved by only the Waldmeisters and SOS.

Table 8

UEQ division results

ATP System	UEQ /100	Avg time	Prfs out
Waldmeister 704	100	2.4	100
Waldmeister 702	94	1.6	94
E-SETHEO csp04	74	21.9	0
Vampire 7.0	73	54.1	73
E 0.82	72	9.3	0
SOS 1.0	58	45.8	58
Otter 3.3	31	17.7	31

5. Descriptions of the Winning Systems

Vampire 7.0 [7], the MIX and FOF divisions winner, is a strategy scheduling ATP system for first-order classical logic. Its kernel implements the calculi of ordered binary resolution with superposition for handling equality. The splitting rule and negative equality splitting are simulated by the introduction of new predicate definitions and dynamic folding of such definitions. A number of standard redundancy criteria and simplification techniques are used for pruning the search space: subsumption, tautology deletion (optionally modulo commutativity), subsumption resolution, rewriting by ordered unit equalities, basicness restrictions, and irreducibility of substitution terms. The reduction orderings used are the standard Knuth-Bendix ordering and a special nonrecursive version of the Knuth-Bendix ordering. A number of efficient indexing techniques are used to implement all major operations on sets of terms and clauses. Run-time algorithm specialization is used to accelerate some costly operations, e.g., checks of ordering constraints. Although the kernel of the system works only with CNF, the preprocessor component accepts a problem in FOF, clausifies it, and performs a number of useful transformations before passing the result to the kernel. When a refutation is found, Vampire produces verifiable output, which validates both the clausification phase and the refutation of the CNF. Since Vampire 6.0 the kernel implementation has undergone a number of significant changes, and several new features has been added. These include implementation of a lightweight basic superposition calculus, better goal-oriented selection of precedence relations on predicates and functions, and some goal-oriented equality reasoning. The automatic mode of Vampire 7.0 is derived from extensive experimental data obtained on problems from TPTP v2.6.0. Input problems are classified taking into account simple syntactic properties, such as being Horn or non-Horn, presence of equality, etc. Additionally, Vampire takes into account the presence of some important kinds of axioms, such as set theory axioms, associativity, and commutativity. Every class of problems is assigned a fixed schedule consisting of a number of kernel strategies (up to 30 strategies in some cases) called one by one with different time limits.

Gandalf c-2.6-SAT, the SAT division Assurance class winner, and Paradox 1.0 [3], the SAT division Model class winner, were the winners in CASC-19, and were described in the CASC-19 report [13].

DCTP 10.21p, the EPR division winner, is a strategy parallel prover using the technology of E-SETHEO [10] to combine several different strategies based on DCTP 1.31 [8,9]. DCTP-10.21p is implemented as a Perl program that briefly analyses the problem and then invokes a schedule of several DCTP strategies. For the EPR division, this schedule consists of merely two of these strategies. An additional strategy had been introduced for "big" EPR problems that are by themselves trivial, but require efficient handling due to their sheer size. However, after fixing a performance bug this additional schedule became redundant. In the EPR division DCTP 10.21p solved all but one of the problems, the unsolved one being an equational variant of the pigeonhole problem. The fact that most of the solutions were found by one particular strategy can be attributed to very good performance of any DCTP strategy on the EPR division: the successful strategy was simply the first one in the strategy schedule.

Waldmeister 704 [5], the UEQ division winner, is a system for unit equational deduction. Its theoretical basis is unfailing completion with refinements towards ordered completion. The system saturates the input axiomatization, distinguishing active facts, which induce a rewrite relation, and passive facts, which are the one-step conclusions of the active ones up to redundancy. The saturation process is parameterized by a reduction ordering and a heuristic assessment of passive facts. Waldmeister 704 features the following improvements over previous versions: First, the detection of redundancies in the presence of associative-commutative operators has been strengthened (cf. [6]). In a set of AC-equivalent equations, an element is redundant if each of its ground instances can be rewritten, with the ground convergent rewrite system for AC, into an instance of another element. Instead of elaborately checking this kind of reducibility explicitly, it can be rephrased in terms of ordering constraints and efficiently approximated with a polynomial test. Second, the last teething troubles of the implementation of the Waldmeister loop have been overcome. Third, a number of strategies have slightly been revised. The prover can be obtained from:

http://www.waldmeister.org.

6. Conclusion

CASC-J2 was the ninth large scale competition for first order ATP systems. Improved performances (relative to the CASC-19 winner) in the MIX division indicates general progress in that area. The divergence between the top systems and the "also-rans" in the MIX divisions, noted in [13], remained salient, and was also observed in the FOF and SAT divisions. The few top systems are the product of deep theoretical and practical knowledge, coupled with significant development effort. While the developers of these top systems reap the rewards of their unique situation, the field as a whole would benefit from a wider range of available state-of-the-art systems. This is an issue that can be addressed through explicit support and recognition for implementation efforts. CASC-J2 benefited from the entry of three new systems, which illustrated the potential of new calculi and strategies. It is hoped that other fledgling developers will realize that the benefits of being a part of CASC far outweigh any perceived disadvantages of not being one of the top few performers.

For CASC-20 it is planned to promote the FOF division to the primary place. This change is motivated by increased use of FOF in applications, and most of the recent contributions to the TPTP have been in FOF format. The failure of the CNF-based systems in the FOF division to cope with the large formulae in the new `ALG` problems indicates a need for, and is expected to stimulate research and development of, better FOF to CNF converters. Techniques that can deal directly with FOF problems may also be improved.

CASC-J2 fulfilled its objectives, by evaluating the relative abilities of current ATP systems, and stimulating development of and interest in ATP systems. The competition highlighted areas of ATP where progress was made in the last year, and through the continuity of the event the results allow performance comparisons with previous and future years. The competition provided exposure for system builders both within and outside of the community, and provided an overview of the implementation state of running, fully automatic, first order ATP systems.

References

[1] P. Baumgartner, A. Fuchs, and C. Tinelli. Darwin - A Theorem Prover for the Model Evolution Calculus. In G. Sutcliffe, S. Schulz, and T. Tammet, editors, *Proceedings of the Workshop on Empirically Successful First Order Reasoning, 2nd International Joint Conference on Automated Reasoning*, Electronic Notes in Theoretical Computer Science, 2004.

[2] P. Baumgartner and C. Tinelli. The Model Evolution Calculus. In F. Baader, editor, *Proceedings of the 19th International Conference on Automated Deduction*, number 2741 in Lecture Notes in Artificial Intelligence, pages 350–364. Springer-Verlag, 2003.

[3] K. Claessen and N. Sorensson. New Techniques that Improve MACE-style Finite Model Finding. In P. Baumgartner and C. Fermueller, editors, *Proceedings of the CADE-19 Workshop: Model Computation - Principles, Algorithms, Applications*, 2003.

[4] S. Colton, A. Meier, V. Sorge, and R. McCasland. Automatic Generation of Classification Theorems for Finite Algebras. In M. Rusinowitch and D. Basin, editors, *Proceedings of the 2nd International Joint Conference on Automated Reasoning*, number 3097 in Lecture Notes in Artificial Intelligence, pages 400–414, 2004.

[5] T. Hillenbrand. Citius altius fortius: Lessons Learned from the Theorem Prover Waldmeister. In I. Dahn and L. Vigneron, editors, *Proceedings of the 4th International Workshop on First-Order Theorem Proving*, number 86.1 in Electronic Notes in Theoretical Computer Science. Elsevier Science, 2003.

[6] B. Loechner. A Redundancy Criterion Based on Ground Reducibility by Ordered Rewriting. In M. Rusinowitch and D. Basin, editors, *Proceedings of the 2nd International Joint Conference on Automated Reasoning*, number 3097 in Lecture Notes in Artificial Intelligence, pages 45–59, 2004.

[7] A. Riazanov and A. Voronkov. The Design and Implementation of Vampire. *AI Communications*, 15(2-3):91–110, 2002.

[8] G. Stenz. DCTP 1.2 - System Abstract. In C. Fermüller and U. Egly, editors, *Proceedings of TABLEAUX 2002: Automated Reasoning with Analytic Tableaux and Related Methods*, number 2381 in Lecture Notes in Artificial Intelligence, pages 335–340. Springer-Verlag, 2002.

[9] G. Stenz. *The Disconnection Calculus*. Logos Verlag, 2002. Dissertation, Fakultät für Informatik, Technische Universität München.

[10] G. Stenz and A. Wolf. E-SETHEO: An Automated Theorem Prover. In R. Dyckhoff, editor, *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX-2000)*, number 1847 in Lecture Notes in Artificial Intelligence, pages 436–440. Springer-Verlag, 2000.

[11] G. Sutcliffe. The CADE-16 ATP System Competition. *Journal of Automated Reasoning*, 24(3):371–396, 2000.

[12] G. Sutcliffe. Proceedings of the 2nd IJCAR's CADE ATP System Competition. Cork, Ireland, 2004.

[13] G. Sutcliffe and C. Suttner. The CADE-19 ATP System Competition. *AI Communications*, 17(3), 2004.

[14] G. Sutcliffe and C.B. Suttner. The TPTP Problem Library: CNF Release v1.2.1. *Journal of Automated Reasoning*, 21(2):177–203, 1998.

[15] C. Weidenbach, U. Brahm, T. Hillenbrand, E. Keen, C. Theobald, and D. Topic. SPASS Version 2.0. In A. Voronkov, editor, *Proceedings of the 18th International Conference on Automated Deduction*, number 2392 in Lecture Notes in Artificial Intelligence, pages 275–279. Springer-Verlag, 2002.