Editorial

CARLOS ARECES, *ILLC, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands. E-mail: carlos@wins.uva.nl*

ENRICO FRANCONI, *IMG, Department of Computer Science, University of Manchester, Oxford Rd., Manchester M13 9PL, UK. E-mail: franconi@cs.man.ac.uk*

RAJEEV GORÉ, *Automated Reasoning Project and Department of Computer Science, Australian National University, Canberra, ACT, 0200, Australia. E-mail: Rajeev.Gore@arp.anu.edu.au*

MAARTEN DE RIJKE, *ILLC, University of Amsterdam, Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands. E-mail: mdr@wins.uva.nl*

HOLGER SCHLINGLOFF, *TZI-BISS, Universität Bremen, P.O. Box 330440, D-28334 HB, Germany. E-mail: hs@tzi.org*

The first Methods for Modalities workshop (M4M-1) was born late at night in November 1998, with a knock on a door, a small but adequate budget, and the urge to do new things. It grew from the good disposition of invited speakers, the enthusiasm of contributors, and long hours behind the computer answering mails, printing articles, organizing time tables, designing web pages and answering phone calls. The aim was clear: to bring together some of the people, logicians and/or computer scientists who were, in one way or another, computing with modal and modal-like logics such as description logic, hybrid logic, temporal logic, etc.

The connections between logic and computing are wide-spread and varied. Well-known examples of uses of logic in computer science include automated verification [25], databases [2], knowledge representation [8], artificial intelligence [20], formal languages [28], etc. Going in the opposite direction, from computer science to logic, we find extremely fast implementations of model checkers and tableaux-based and resolution-based theorem provers [9], automata-theoretic methods for deciding powerful languages [5], tight connections between the theories of computational and descriptive complexity [24], etc. And this is just a small part of a far bigger development, as logic continues to play an important role in computer science and permeating more and more of its main areas. All signs indicate that computer science and logic have decided to establish a stronghold together and profit from the interchange of ideas. This development has been recognized throughout the community, as is witnessed, for instance, by this year's launch of the *ACM Transactions on Computational Logic* [30], the founding of IFCOLOG, the International Federation on Computational Logic [23], and the first installment of the International Conference on Computational Logic [12].

While the links between computer science and *modal* logic may be viewed as nothing more than specific instances of these developments, there is something special to them. Graphs are the key. Graphs are ubiquitous in computer science: think of transition systems, parse trees, Petri nets, decision diagrams, flow charts, ... It is because of this, that modal languages are so well suited to describe computer science phenomena: Kripke models, the standard semantic structure on which modal languages are interpreted, are nothing but graphs [4]. Of course, graphs, or more generally relational structures, are also the semantic structures of choice for other languages, including first-order logic, but from a computational point of view, modal languages have a *huge* advantage when compared to first-order logic: they are usually decidable.

The first workshop on Method for Modalities focused on the particular relationship between modal formalisms and computer science. While planning M4M-1, we aimed to create an environment in which one could really *learn* from what the speakers coming from different fields would have to tell us. We settled on long, tutorial-like presentations by invited speakers, complemented by contributed papers and an afternoon for system demonstrations.

The invited talks covered some of the main approaches to modern automated reasoning (resolution and tableaux calculi), discussed techniques for test set generation, model checking and labeled deduction. Submitted papers, on the other hand, contributed further, less traditional angles. System presentations included both classical and state of the art modal theorem provers, demonstrated by the developers themselves, which helped to create a genuine "hands-on" atmosphere.

The workshop proved to be a stimulating event that motivated people to share knowledge and expertise. Lecture rooms overflowed as we underestimated the popularity of our late-night-idea. And the ripples created by the pebble we threw back in November 1998 are still moving out from the center. Material related to the workshop is available online at the M4M web site (`http://www.illc.uva.nl/~m4m/`); this should serve as an "entry point" for anyone interested in exploring the field. And, of course, M4M-2 is being planned while we write this editorial.

M4M-1 in this Special Issue

Some of the speakers taking part in the workshop were invited to prepare journal versions of their presentation for publication in this special issue. We also invited some of the speakers to join up to form "teams" and provide us with a broad view of their area of expertise. We are extremely glad that the authors concerned took the opportunity and agreed to team up and collaborate.

After a formal reviewing process, five papers were selected as a fair representation of the material presented at M4M-1. Of those five, three focus on methods and methodologies for traditional modal and modal-like formalisms, while two aim to extend familiar views of such formalisms to more inclusive ones, either in terms of general fragments of classical logics or in terms of sorting and naming mechanisms. Let us briefly introduce each of the papers.

Practical Reasoning for Very Expressive Description Logics by Ian Horrocks, Ulrike Sattler, and Stephan Tobies.

Description logics are a family of languages especially devised for knowledge representation [16]. The connection between description logics and modal logics has a long history. The first results date back to [29]; these were extended in, for example, [15].

In application areas such as knowledge representation, one may need very expressive description logics. For instance, one may need to be able to deal with converse relations, number restrictions, A-Box and T-Box reasoning, transitivity, etc. Even though the worst case complexity of the satisfiability problems for these languages is usually EXPTIME, the known algorithms have a good average case performance.

The paper describes one of the most expressive families of description logics nowadays: $\mathcal{ALC}$ extended with transitive roles, hierarchies and converse. It explores the differences between transitive roles and taking the transitive closure of roles, with respect to tableau-based decision algorithms. It establishes PSPACE-completeness of the satisfiability problem for $\mathcal{SI}$, while an undecidability result is obtained when unrestricted number quantification is also allowed. Finally, the paper discusses various optimization techniques.

The salient characteristic of the paper is its mixture of theoretical and application driven results, which is almost a hallmark of the literature on description logics.

Resolution-Based Methods for Modal Logics by Hans de Nivelle, Renate Schmidt, and Ullrich Hustadt.

Automated theorem proving for first-order logics is a well developed field with years of history [9]. Attending a conference on the field, like for example CADE, International Conference on Automated Deduction, can be an unforgettable experience, where highly tuned heuristics are discussed and test-beds are crunched down by provers trying to outperform each other.

Now, the standard or relational translation [32] embeds modal logics into first-order logics while preserving satisfiability; hence, it opens the door to modal theorem proving by means of first-order techniques. The power of first-order logic translates into both advantages and disadvantages from the modal point of view. On the one hand, it allows one to explore combinations of logics, as different logics can be *jointly* translated into first-order logic; the resulting theory can then be fed to a first-order prover. But, of course, there is the issue of (un-)decidability.

The latter is the topic of the paper, which focuses primarily on first-order resolution methods and how to turn them into decision methods for modal logics. The authors start by introducing the general resolution framework and then go on to discuss refinements of resolution by means of orders and selection functions which are powerful enough to "tame" the method and transform it into a decision procedure for a variety of modal logics.

To complete the picture, the authors also draw connections between resolution and tableau based modal theorem proving, and comment on simulation results.

An Analysis of Empirical Testing for Modal Decision Procedures by Ian Horrocks, Peter F. Patel-Schneider, and Roberto Sebastiani.

It is generally accepted that the worst case complexity of the satisfiability problem for a modal language is not a fair measure to assess the typical case complexity. Many algorithms deciding the same language might live in the same complexity class but perform completely differently when tested on randomly generated or real life problems. Usually, empirical testing is the only way to assess typical case complexity.

The area of testing in propositional logic is well developed and the easy-hard-easy pattern of propositional satisfiability well studied [6]. But little is know about testing for modal logics, which poses a much more complex challenge. For a start, decidable modal languages usually live in one of three different complexity classes: the very easy ones like **S5** in NP, the classical ones in PSPACE like **K** or **T**, and the hard ones like PDL in EXPTIME; and different tests are needed for each of these classes. In addition, the typical behavior of modal logics in those classes need not be the same.

The paper provides a survey of empirical testing methodologies and analyses the current state of the art, proposing criteria for defining a "good test set." In addition, a new test generation methodology is proposed as a variation of the $3\mathrm{CNF}_{\Box_m}$ algorithm.

Reachability Logic: An Efficient Fragment of Transitive Closure Logic by Natasha Alechina and Neil Immerman.

As we said, classical modal languages can be embedded into first-order logic by means of the standard translation mentioned above. But certain extended modal languages, such as PDL or CTL* [21], allow the Kleene star operator and hence are beyond first-order logic. FO(TC), first-order logic extended with transitive closure [17], is a well studied logic in the field of descriptive complexity. The aim of the paper is to identify fragments of FO(TC) that are general enough to "cover" modal logics like PDL and CTL* but still restricted enough to admit model checking algorithms of very low (linear) complexity.

In particular, the paper introduces a 'modal' fragment of $\mathrm{FO}^2(\mathrm{TC})$, the language FO(TC) with only two variables, where quantifiers are restricted by path descriptions, which is later extended with boolean variables. The model checking algorithm for this fragment, which is called $\mathcal{RL}$, is linear in the size of the formula and the size of the model, but exponential in the number of boolean variables. Interestingly, the boolean variables play an important role. The authors show that neither PDL nor CTL* can be embedded in $\mathcal{RL}$ without the use of boolean variables. Furthermore, the number of boolean variables are a good indicator of the complexity of a given CTL* query.

Representation, Reasoning, and Relational Structures: a Hybrid Logic Manifesto by Patrick Blackburn.

This paper views modal logics as languages that are especially tailored for describing relational structures. After pointing out the connections between modal logics and other fields, it describes a natural extension to traditional modal languages: hybrid logics.

Hybrid logics are languages with the ability to explicitly refer to states in a model. This capacity, which is absent in traditional modal languages, makes hybrid languages especially well-suited for many modeling tasks, including knowledge representation, the analysis of linguistic phenomena, and temporal reasoning. Blackburn starts by discussing simple hybrid logics where only names and satisfaction operators are added. But he also gives a taste of more expressive languages where names can be bound globally or locally, new sorts are introduced, etc.

Blackburn manages to get many important intuitions across in a very accessible manner. Examples abound in the paper, which will help the reader to pull together the many threads from logic, linguistics and computer science that are present in the paper.

M4M-1 not in this Special Issue

For a variety of reasons there is a number of very interesting topics that were covered at M4M-1 but that did not make it into this special issue. In this section we briefly comment on each of them.

Logic offers the possibility of modeling and reasoning about hardware and software systems. But which logic? In his presentation, Basin [3] proposed monadic logics of strings and trees as good candidates for many kinds of discrete systems. The connection between such logics and modal logics is established at the level of frames, and the decidability result of Rabin [27] for SnS has often been used to prove decidability of modal systems by embeddings.

The use of modal and modal-like languages as modeling tools was illustrated in two presentations. Bleeker and Meertens [7] reported on work dealing with modal logics able to capture the dynamics of knowledge during communication, with a view to understanding security protocols. Van Eijck, de Boer, van der Hoek and Meyer [33] presented work on a modal logic with a special kind of quantification, aimed at modeling network topologies.

Traditional proof-theoretical concerns were also represented at M4M-1, where Governatori and Rotolo [22] discussed their recent work on modal proof theory in the spirit of Gabbay's labeled deductive systems, Fariñas del Cerro and Gasquet [14] presented tableaux-based decision procedures for modal logics of confluence and density, and Ohlbach [26] introduced a new theory resolution style calculus for combined A-Box and T-Box reasoning.

Finally, Cerrito, Mayer and Praud [13] presented work on first-order linear time temporal logics over finite frames; in particular, they discussed a number of results on undecidability and high undecidability.

The following systems were demonstrated at M4M-1: $\Box$-KE, developed by Cunningham, Pitt, Williams and Kamara; Akka, developed by Hendriks; FaCT and iFaCT developed by Horrocks; lc2 developed by Marx and Schlobach; Bliksem, developed by de Nivelle; and DLP, developed by Patel-Schneider.

While the scope of M4M-1 was broad and while we managed to attract contributions on a wide variety of topics, it was only a two-day event: various important topics on the interface of modal logic and computing were not addressed during M4M-1, thus suggesting obvious topics for M4M-2 and other future installments of the workshop.

As regards the application of modal logic to hard-core computer science, the work on system verification (like temporal languages for real time systems or the recent advances in model checking [11]) was barely present. Logic programming, and its modal extension to knowledge programming [18], is also a clear topic for M4M, as is, more generally, the connection between modal logics and databases, both at the modeling and inference level. As far as decision methods for modal logics is concerned, there were two important absentees at M4M-1: sequent calculi [34] and automata-based techniques [31]. As we mentioned, modal languages are used as modeling tools in very diverse areas, including computational linguistics, information retrieval, natural language semantics, system design, ... — it would be good to have a fair amount of case studies from as large a subset of these disciplines as possible. Finally, the issue of transfer results for combinations of logics deserves attention [19], as do new connections linking modal languages to game theory [10].

As we tried to convey with the title of this editorial introduction, the message we want to get across is *Use Your Logic*. And we mean *use* in a very concrete way. Modal logics provide restricted yet expressive languages for modeling a wide variety of problems, and today we have automated tools that can perform modal inference with ever increasing efficiency. "Make your life easier, have a theorem prover installed," might be the slogan if an advertising agency were behind M4M. But the phrase is not a mere slogan for us.

Program Committee

The program committee for M4M-1 consisted of Carlos Areces (Amsterdam), Enrico Franconi (Manchester), Rajeev Goré (Canberra), Hans de Nivelle (Saarbrücken), Hans Jürgen Ohlbach (London), Maarten de Rijke (Amsterdam), as well as Holger Schlingloff (Bremen).

Out of 20 submissions, the program committee selected 9 papers for presentation at the workshop. In addition, David Basin, Patrick Blackburn, Ian Horrocks, Hans de Nivelle, Renate Schmidt, and Roberto Sebastiani were invited to address the meeting.

Referees

The program committee gratefully acknowledges the help of the following referees: Guiseppe De Giacomo, Chiara Ghidini, Christoph Lüth, Fabio Massacci, Stephan

Merz, Till Mossakowsk, Markus Roggenbach, George Russell, and Heinrich Wansing.

Sponsors

M4M-1 was sponsored by the Computational Logic group at the Faculty of Science of the University of Amsterdam; the Institute for Logic, Language and Computation (ILLC); Chinaski WorldWide; and the Netherlands Organization for Scientific Research (NWO). We gratefully acknowledge their generous support.

Acknowledgments

We would like to thank Rafael Accorsi, Rosella Gennari, and Marco de Vries for practical assistance before and during M4M-1. We are also grateful to the editors of the *Logic Journal of the IGPL* for their enthusiasm and guidance.

References

[1] C. Areces and M. de Rijke, editors. *Workshop Proceedings Methods for Modalities 1*, Amsterdam, May 1999. ILLC, University of Amsterdam.

[2] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, Reading, MA, 1995.

[3] D. Basin. Verification based on monadic logic. In Areces and de Rijke [1].

[4] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2000. To appear.

[5] D. Basin and N. Klarlund. Automata based symbolic reasoning in hardware verification. *Formal Methods in Systems Design*, 13:255–288, 1998.

[6] H. Kleine Büning and T. Lettman. *Propositional Logic: Deduction and Algorithms*. Cambridge University Press, 1999. Originally published in German by B. G. Teubner, 1994.

[7] A. Bleeker and L. Meertens. Knowledge in security protocols: an operational semantics for BAN logic. In Areces and de Rijke [1].

[8] G. Brewka, editor. *Principles of Knowledge Representation*. CSLI Publications, 1996.

[9] W. Bibel and P. Schmitt, editors. *Automated Deduction – A Basis for Applications*. Kluwer Academic Publishers, 1998.

[10] G. Bonanno and W. van der Hoek. *LOFT-3*. Special issue of *Games and Economic Theory*, 2000.

[11] E. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 2000.

[12] CL2000: First International Conference on Computational Logic. URL: `http://www.doc.ic.ac.uk/cl2000/`.

[13] S. Cerrito, M. Mayer, and S. Praud. First-order linear time temporal logic over finite time frames is not semi-decidable. In Areces and de Rijke [1].

[14] L. Fariñas del Cerro and O. Gasquet. Tableaux based decision procedures for modal logics of confluence and density. In Areces and de Rijke [1].

[15] G. De Giacomo and M. Lenzerini. Boosting the correspondence between description logics and propositional dynamic logics. In *Proc. 12th National Conference on Artificial Intelligence (AAAI'94)*, pages 205–212, 1994.

[16] F. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics. In G. Brewka, editor, *Principles of Knowledge Representation*, pages 191–236. CSLI Publications, Stanford, 1996.

[17] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer, 1995.

[18] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge*. The MIT Press, 1995.

[19] D. Gabbay and M. de Rijke, editors. *Frontiers of Combining Systems 2*. Research Studies Press/Wiley, 2000.

[20] D. Gabbay, C. Hogger, and J. Robinson, editors. *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 1. Oxford University Press, Oxford, 1993.

[21] R. Goldblatt. *Logics of Time and Computation*. CSLI Publications, 1992.

[22] G. Governatori and A. Rotolo. Labelled modal sequents. In Areces and de Rijke [1].

[23] IFCOLOG: International Federation for Computational Logic. URL: `http://www.compulog.-org/net/IFCoLog.html`.

[24] N. Immerman. *Descriptive Complexity*. Springer, 1999.

[25] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems. Vol. 1, Specification*. Springer-Verlag, New York, 1992.

[26] H. Ohlbach. A theory resolution style A-Box calculus. In Areces and de Rijke [1].

[27] M. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969.

[28] G. Rozenberg and A. Salomaa. *Handbook of Formal Languages*. Springer, 1997.

[29] K. Schild. A correspondence theory for terminological logics. In *Proc. of the 12th IJCAI*, pages 466–471, 1991.

[30] ACM Transactions on Computational Logic. URL: `http://www.acm.org/tocl/`.

[31] M. Vardi. Why is modal logic so robustly decidable? In *DIMACS Ser. Disc. Math. Theoret. Comp. Sci. 31*, pages 149–184. AMS, 1997.

[32] J. van Benthem. Correspondence theory. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic. Vol. II*, volume 165 of *Synthese Library*, pages 167–247. D. Reidel Publishing Co., Dordrecht, 1984.

[33] R. van Eijk, F. de Boer, W. van der Hoek, and J. Meyer. A modal interpretation of quantification. In Areces and de Rijke [1].

[34] H. Wansing. *Displaying Modal Logic*. Kluwer, 1998.

Received February 29, 2000