

Editorial

Welcome to the 5th issue of the newsletter. As you can see the newsletter has a new layout in anticipation of its evolution into the magazine "Communications of Applied Logic" to be released sometime in the first half of 2004. We will be happy to receive your comments on this new outlook of the newsletter.

In this issue you will find information about new projects with strong links to computational logic to start soon, activities of the network related to European Enlargement and educational programs. This issue contains also two short review articles in the area of formal methods.

The network is looking to enlarge its scope and tighten its links internationally. One new area it is currently studying of including is that of Logic and Law. You will find a short position piece on this possibility in this issue of the newsletter. ❖

Antonis Kakas and Marinos Georgiades
University of Cyprus

Recent Trends in Computer-Aided Verification

Formal Methods, Specification and Verification

Roderick Bloem
Institute for Software Technology, TU Graz

Introduction

Model checking has generated serious interest over the last decade. Several key findings, including symbolic model checking and abstraction, have enabled the migration of formal verification techniques from the academic to the industrial sector. Many large commercial companies now use in-house formal verification tools, and several commercial offerings exist.

Model checking in its purest form means automated verification of finite-state systems, using a specification in a temporal logic. The term is used more loosely, though, to include verification with some user interaction, verification of infinite-state systems, and verification using other formalisms than a temporal logic. Model checking is rigorous and light-weight. In contrast to refinement-based systems, model checking allows the user to specify and verify only the properties of interest.

This paper surveys trends in model checking that have occurred over the last five years. A survey of recent trends necessarily reflects the author's interests. This short paper focuses on two topics: SAT-based verification techniques and model checking of software. Other, equally important trends have had to be ignored.

Sat-based Verification

Initial approaches to model checking used explicit state enumeration, in which every state in the transition graph is represented and inspected individually.

Though this approach is still in use, it is infeasible for

INSIDE THIS ISSUE

- 1 Editorial
- 1 Recent Trends in Computer-Aided Verification
- 2 Executive Council Report
- 8 A New FP6 Network of Excellence
- 10 Searching for Researchers ...
- 11 Implementing Rational Features ...
- 17 Formal Requirements Engineering...

Continued on page 5

ABOUT THE NEW DESIGN OF THE NEWSLETTER

This document was created using linked text boxes, which allow articles to flow continuously across pages. For example, the article on the right of page one continues on page five.

We also use sidebar articles for any information we want to keep separate from other articles or information that highlights an article next to it. These could include a list of contributors, addresses or contact information, a smaller self-contained story, announcements, a preview of the next issue, or a calendar of schedule.

PICTURES FROM PADOVA MEETING



Executive Council Report

Heike Scheuerpflug

Over the past seven months CoLogNET has been pushing forward with an ambitious spectrum of activities all designed

- to consolidate and enhance the networks education & training activities
- to improve our website facilities
- to increase awareness about the network & its activities via promotional material
- to further our links with industry and to promote our role in technology transfer.

In the following we present a brief overview of the recent developments.

Executive Council Meeting

With the excellent co-organisation and support of Francesca Rossi the forth Executive Council was held on 17 and 18 November 2003 in Padua. All council members attended the meeting. As usual the meeting proved to be vivid, controversial and constructive in discussions and talks. All council members are highly engaged in the network and draw their motivation to commit a follow-up proposal from their achievements in the running network. The presentations held on 17 November to report on the progress within each workpackage and area showed that the network continues its path to take the lead in computational logic in Europe and the world. On 18 November we initiated first discussions on a successor project of CoLogNET within the 6th framework programme and the different options that would be open for the novelty objectives within a follow-up project. The network will apply for a no-cost extension to complete the final report and prepare for the new project. Proposal submission will be coordinated from Saarbrücken, DFKI GmbH Saarbrücken, in cooperation with the future coordinator. Jörg Siekmann will not be available to run for office as the project coordinator in the successor project, however, he will continue to support the new initiative as much as possible. The discussion about the new & traditional instruments in FP6 and goals that a successor project should accomplish must be further consolidated and an agreement must be reached

among the consortium on who will become the coordinator in the future project. In addition a workshop to align the integration activities of the area websites for the Who's Who infrastructure was held in the afternoon of 18 November 2003.

CoLogNET Portal

In accordance with the review report and the reviewer who called for some slight changes of the CoLogNET portal to emphasise better the academic subject "Computational Logic" instead of the networks managerial aspects we regrouped the different sections. A new section MISSION was set up to show the vision of CoLogNET. All managerial aspects related to the network and structure of the network have been merged into the section NETWORK STRUCTURE, which provides detailed information on Management Structure, Executive Council, Task Forces and Members. It was also recommended to better link the area websites into the main portal. This activity is also making progress and the information exchange protocol has already been implemented between the area websites of CLP and ITCLS. In a next step the protocol will also be implemented between the area websites of CLP, ITCLS, NLP and Who's Who. Eventually, the final protocol will also be implemented between the Who's Who and the main portal. Most area websites have adapted the common look and feel of the main CoLogNET website to be in line with CoLogNET's corporate identity.

Who is Who in Logic

One of the objectives of the network is to set up a large and international Who's Who in logic in the world, to establish a main portal for the Who's Who repository, including researchers' profiles, publications & main events. The development of a content management system is in full progress and a working prototype based on Zope/Plone is in place. The system will work with sites communicating directly with the Who's Who, but in the initial phase not directly with each other. Information between sites can be shared through the Who's Who.

Education & Training

A complete description of courses within the distributed masters programme including course material is available online. Furthermore, the website for open

positions & grants as well as the forum to discuss PhD-proposals have been set up. Colloquia are broadcasted from Dresden to the CoLogNET partner sites and the International Centre for Computational Logic (ICCL) has been founded at TU Dresden. The education & training group is also furthering its activities in the dual master degree programme and a joint PhD programme between Dresden and Lisbon, and the European Masters in Computational Logic. The Formal Methods area of CoLogNET headed by Dines Bjørner is organising a PhD Summer School in June 2003. The purpose is to help educate and train on the highest level some 40-50 PhD students in the area of logics and formal software specification languages. Plans are also afoot to organise an international symposium on Teaching Formal Methods. Further information is available at <http://www.imm.dtu.dk/%7Edb/colognet/index.php?page=teaching>. Since October 2003 Folli, the European Association for Logic, Language and Information is coordinating a task related to the education & training activities within CoLogNET. The ambitious aim is to produce a "Living Book" by drawing from material produced during the European Summer School in Logic, Language and Information held every year and to create dynamic teaching material based on computational logic tools. More details at <http://www.folli.uva.nl/Projects/Colognet/index2.htm>

Technology Transfer

The networks most ambitious venture is now underway: The launching of ForTIA – the Formal Techniques Industrial Association at FME'03 in Pisa. ForTIA is an association of industrial companies, comprising both suppliers and, above all, users, of formal techniques. Its aim is not just one of mutual benefit both for industrial and academic partners. It is also about information sharing and active contributions to ensure that good tools and techniques are researched, developed and deployed. Also Task Force 2 promoting the links between CLP and industry has been very successful. A one day ECLIPSE school has been organised and 9 demo sessions have been held at CP 2003 in Kinsale. The NLP group of Michael Moorgat is organising a industrial event in December at the Amsterdam Colloquium. Werner Ceuster will give his invited talk on Language and Computation.

Relations with Eastern European Countries

A new scheme was set up in compliance with the rules set out in the Marie Curie Actions – Human Resources and Mobility Activity. The main aim of this action is to provide partial financial support for paid stays at a host organization within the CoLogNET network. The role of this activity is to support research training and exchange of knowledge by providing financial means for researchers from Associated States. There are two main categories of researchers eligible for funding:

-Early-stage researchers: This refers to researchers at the beginning of their research career with less than four years' active research experience (e.g. researchers undertaking a doctoral degree);

-Experienced researchers: This applies to researchers with more than four years of active research experience or those with a doctorate degree. This activity will not be eligible for researchers with more than ten years of experience. New Area "Logic and Law" A new Area Logic and Law was proposed to replace the original WP10. In accordance with the review report a revised work package description and budget distribution was sent to the Executive Council and the Commission for final approval. The Executive Council and the Commission agreed to integrate the new area Logic and Law in the still pending amendment.

Relations with other projects and networks

-Relations with KTweb

We have further strengthened our links with KT Web by distributing the CoLogNET newsletter to the KTweb community and we also strongly encourage CoLogNET members to take advantage of the communication platform and services KTweb offers. It is possible to submit articles, fact sheets, news, events & links. For more information visit the Ktweb portal at www.ktweb.org.

- Relations with FoLLi

In addition we have fostered strong relations with the European Association for Logic, Language and Information (FoLLi) and the European Summer School in Logic, Language and Information (ESSLLI). In

October 2003 Folli with the support of CoLogNET has started coordinating a task on E-learning in Computational Logic – called the Living Book. A website has been set up and further information is available at <http://www.folli.uva.nl/Projects/Colognet/index2.htm>

-Relations with IFCoLog

The executive council of CoLogNET has approved to set up a parallel news magazine entitled Applied Logic Communications - The official magazine of the International Federation of Computational Logic which will eventually merge with the CoLogNET newsletter. With the strong support of CoLogNET as the main driving force for and within IFCoLog, IFCoLog will be registered as a charity in the UK with offices at King's College London. The Charity allows for an enormous tax advantage for donations, fund raising and income. The IFCoLog website has been set up and is in full operation at <http://www.colognet.org/IFCoLog/>

Promotional Drive

To increase general awareness of CoLogNET's activities in research, technology transfer and education and training and to promote the network's infrastructure facilities we have produced a rather explicit flyer and a membership application form which can be downloaded from the web or ordered from DFKI GmbH Saarbrücken. In addition we produced a new set of posters which were integrated in the mobile booth. Upon request DFKI GmbH Saarbrücken sends dissemination material including flyers, applications forms and mobile booth to the CoLogNET partner sites.

Summary & Outlook

By the end of December 03 CoLogNET will have already accomplished two years of its project life cycle and is having a significant effect on European and world research. We believe that we met the recommendations set out in the review report. The next review meeting will be held in form of a joint cluster meeting on 26 January in Brussels. The amendment process will be finalised by the end of December this year and we expect to receive the amendment contract in time for the next reporting period which will start in January. By the mid of 2004 the network will start to prepare for its successor project. First discussions on the future of CoLogNET have been initiated and need to be consolidated in further discussions and negotiations with the Commission and the CoLogNET consortium. ❖

CALENDAR OF EVENTS

2ND COLOGNET-ELSNET SYMPOSIUM

AMSTERDAM, NETHERLANDS

2003, DECEMBER 18

The symposium is the second of three devoted to the exploration of the common ground between the "Logic and Natural Language Processing" Area of CoLogNET (Network of Excellence in Computational Logic) and ELSNET (Network of Excellence in Human Language Technologies)

14TH AMSTERDAM COLLOQUIUM

AMSTERDAM, NETHERLANDS

2003, DECEMBER 19-21

The Amsterdam Colloquia aim at bringing together linguists, philosophers, logicians and computer scientists who share an interest in the formal study of the semantics of natural and formal languages. The spectrum of topics covered ranges from descriptive (semantic analyses of all kinds of expressions) to theoretical (logical and computational properties of semantic theories, philosophical foundations). (<http://www.uilots.let.uu.nl/~ctl/workshops/CES03/>)

IJCAR 2004 - SECOND INTERNATIONAL JOINT CONFERENCE ON AUTOMATED REASONING

UNIVERSITY COLLEGE CORK, IRELAND

2004, JULY 4-8

SUBMISSION DATE: JANUARY 5TH 2004

IJCAR 2004 is the Second International Joint Conference on Automated Reasoning (IJCAR) and is to be held in Cork, Ireland from July 4th to 8th, 2004. IJCAR will be a merger of CADE, FTP, TABLEAUX, FroCoS (Workshop on Frontiers of Combining Systems) and CALCULEMUS. Satellite workshops, tutorials and co-located events are expected.

A JOINT COLOGNET/FME EVENT: A PHD SUMMER SCHOOL: THE LOGICS OF FORMAL SOFTWARE SPECIFICATION LANGUAGES

THE HIGH TATRAS, SLOVAKIA

2004, JULY 6-9

The discussion topics of the event include (1) A PhD Summer School (2) Plans for a PhD Summer School (3) A CAI Double Issue: Vol.22, Nos.2-3 (<http://www.imm.dtu.dk/~db/colognet/>)

ICFEM '04 - INTERNATIONAL CONFERENCE ON FORMAL ENGINEERING METHODS

SEATTLE, WA, USA

2004, NOVEMBER 8-12

Continued on page 10

very large systems, and a set-based, symbolic method is used instead. In a symbolic approach, sets of states are represented by their characteristic function. In contrast to explicit representations, the size of a symbolic representation of a set is only loosely related to its size.

The semantics of Computation Tree Logic (CTL), an important specification formalism, are usually given in the form of point expressions over sets of states [CE81]. Such expressions over sets are naturally mapped to a symbolic approach using, for example, BDDs [McM94, Bry86].

BDDs provide a canonical representation of sets and relatively efficient versions of the operations needed to compute the CTL formulas: set union and intersection, negation, and existential quantification. The latter is crucial in the computation of the direct predecessors of a set of states, an important operation in evaluating CTL formulas. For synchronous finite state systems, symbolic model checking with BDDs is typically far more efficient than model checking with explicit state enumeration, and it may be said that BDDs are the single most important factor in moving model checking out of the purely academic domain, and into the industrial domain.

Naturally, BDDs have their drawbacks, too. Any representation of a set of states necessarily has a worst case linear complexity in the number of states, or an exponential complexity in the number of bits encoding the state. For BDDs, it is notoriously hard to predict when the worst case will occur, and subsequently, how hard it is to model check a given design. Furthermore, the Tarski-Knaster evaluation order leads to breadth first searches in the state space. The iterations found along the way need to be represented, which may be hard. Some approaches have been devised that deviate from this breadth-first search [RS95, RS99, BRS00], and these approaches are more efficient than pure breadth-first search, though iterates still have to be stored.

Bounded Model Checking

In 1999, a method for symbolic model checking using SAT solvers was proposed [BCCZ99]. The method works for universal and existential fragments of the temporal logics. In its simplest variant, it computes

invariants by unrolling the transition relation, constructing a propositional formula that states that a state violating the invariant can be reached in k or fewer steps, for a fixed k . This formula is then checked for satisfiability by a satisfiability (SAT) solver. Liveness properties are checked in a similar manner, by introducing a loop on the path and checking that no state on the path satisfies the liveness condition.

It is possible, though typically very expensive to compute the sequential depth or the diameter of the circuit, giving an upper bound on the needed value of k . This means that in practice SAT-based model checking is incomplete, because it checks for errors only up to a given depth. Hence its name, Bounded Model Checking (BMC).

BMC is generally seen as more efficient than BDD-based model checking. It does not need to represent iterates, and the maximum length on the possible counterexample allows it to avoid quantification. This, and recent improvements in SAT solving, lead to algorithms that are often more memory efficient.

BMC has gained popularity mainly due to its efficiency, regardless of its incompleteness. Especially in industry SAT-based model checking is valued for its ability to find bugs in large designs, even if it may fail to find all of them. On top of that, BMC does improve confidence in the design by excluding bugs up to a certain depth.

A Complete SAT-Based Method

In a 2003 paper [McM03], McMillan proposes a complete algorithm for model checking based on SAT solving. McMillan's technique uses the BMC proof that the negation of the invariant (the "bad states") can not be reached from a given set of states in k steps or less for a fixed k . From this proof, one can derive a set of states that includes all states that can be reached from the given set in one step, but no states that can reach the bad states in k steps or less.

The algorithm consists of two nested loops. The outer loop controls the value of k , which starts at 1 and increases. For a given value of k , the algorithm tries to prove that a bad state can be reached in k steps. If the proof succeeds, the invariant does not hold. If the proof fails, the second loop, for increasing i , constructs a set R_i of states that includes the states that can be reached from the initial states in i steps, but that does

not include any states that can reach a bad state in k steps or less. If the R_i converge, it constitutes a proof that no bad state can be reached from the initial states. Instead of converging, for some i we may find a bad state that can be reached from R_i . This does not prove that there is a path from the initial states to the bad states, so the algorithm increases k and reiterates.

McMillan's method can prove that an error does not occur in a design, something that BMC can not do. The algorithm uses multiple BMC runs, but it may conclude with a small value of k , with the conclusion that the property holds, while a manual application of BMC would imply the use of large k to gain confidence at least up to that level.

Model Checking Software

The traditional area of model checking is that of finite-state systems. Recently, automated methods of ensuring correctness of software have received increased interest. Here, we will concentrate on two of them, the SLAM project [BR01] and Java Path Finder [BHPV00], ignoring some other important developments.

Model checking software is obviously undecidable, and hence none of the model checking approaches is guaranteed to terminate. Nevertheless, there are many programs that are amenable to model checking, and the tools terminate on these programs with a counterexample or an assurance that a stated property holds.

The SLAM project aims to model check invariants in single-threaded C programs. SLAM was developed at Microsoft Research to certify third-party device drivers, which run in kernel space and can thus wreak havoc if not implemented carefully. Properties are basically stated as assert statements in the code. SLAM uses an abstraction/refinement scheme based on predicate abstraction. If dynamic memory allocation is not used, abstraction results in a "Boolean program" with potentially unbounded recursion, the same control constructs as C, but only Boolean variables. A Boolean program is equivalent to a push-down automaton and, though it has infinite state, it can be checked by a model checker such as Bebop [BR00] or Moped [ES01]. Pointer aliasing is handled by an independent

conservative aliasing algorithm.

The SLAM toolkit works with a set of predicates stating relations between variables in the C program. These predicates correspond to Boolean variables in the Boolean program. The statements in the Boolean program are abstractions of the statements in the C program, approximating the effect of the C statement on the predicates. Statements, including control statements, are abstracted in a conservative manner. When not enough information is available, a statement is allowed to do more rather than less, which may lead to false alarms.

The SLAM toolkit starts with a small set of predicates. If the model checker does not find an error in the corresponding Boolean program, the original C program is guaranteed to be correct. Otherwise, the path to the error is analyzed to see if it exists in the original C program. If so, the program contains an error. If not, the spurious counterexample suggests a set of predicates to be added to the abstraction, making it more precise, after which the process is repeated. Finding a new set of predicates and constructing an abstraction involves heavy use of a theorem prover. SLAM may not terminate either because the theorem prover does not, or because its predicate refinement loop runs without end. The researchers report that they routinely apply the model checking algorithm to device drivers, and that the model checker usually terminates in only a few iterations.

Java Path finder (JPF), in contrast, checks multi-threaded Java programs. JPF can detect errors indicated by assert statements, as well as find deadlocks in concurrent programs. It also includes algorithms for finding possible race conditions.

JPF is based on a homegrown Java virtual machine, which stores a history of states of the Java program. This allows it to backtrack and try different alternatives. By doing a depth-first search, JPF can inspect the entire state space of programs that have limited memory use. If the state space of a Java program is unbounded, JPF may not terminate.

JPF includes predicate abstraction techniques like the ones in the SLAM toolkit, but it does not automatically construct and refine abstractions.

Conclusions

We have described two current trends in formal verification. SAT solving algorithms have helped boost the capacity of model checkers, which is of primary importance to its acceptance. Using recently developed techniques, SAT-based techniques can prove correctness as well as faultiness of designs. Model checking of software is still in its infancy, but promising developments are taking place. Both SLAM and JPF check programs written in every-day programming languages, and SLAM is being used routinely in an industrial setting. Both tools exhibit the typical benefits of model checking: automatic or semi-automatic verification, high confidence in the correctness of the stated properties, and no need for a full formal specification of the desired behavior. ❖

References

- [BCCZ99] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu. Symbolic model checking without BDDs. In Fifth International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'99), pages 193{207, Amsterdam, The Netherlands, March 1999. LNCS 1579.
- [BHPV00] G. Brat, K. Havelund, S. Park, and W. Visser. Java PathFinder Second generation of a java model checker. In Workshop on Advances in Verification, 2000.
- [BR00] T. Ball and S. K. Rajamani. Bebop: A symbolic model checker for Boolean programs. In SPIN 00: SPIN Workshop, pages 113{130. Springer-Verlag, 2000. LNCS 1885.
- [BR01] T. Ball and S. K. Rajamani. Automatically validating temporal safety properties of interfaces. In M.B. Dwyer, editor, 8th International SPIN Workshop, pages 103{122, Toronto, May 2001. Springer Verlag. LNCS 2057.
- [BRS00] R. Bloem, K. Ravi, and F. Somenzi. Symbolic guided search for CTL model checking. In Proceedings of the Design Automation Conference, pages 29{34, Los Angeles, CA, June 2000.
- [Bry86] R. E. Bryant. Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers, C-35(8):677{691, August 1986.
- [CE81] E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. In Proceedings Workshop on Logics of Programs, pages 52{71, Berlin, 1981. Springer-Verlag. LNCS 131.
- [CES86] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite state concurrent systems using temporal logic specifications. ACM Transaction on Programming Languages and Systems, 8(2):244{263, 1986.
- [ES01] J. Esparza and S. Schwoon. A bdd-based model checker for recursive programs. In Thirteenth Conference on Computer-Aided Verification (CAV '01), pages 324{336. Springer-Verlag, 2001.
- [McM94] K. L. McMillan. Symbolic Model Checking. Kluwer Academic Publishers, Boston, MA, 1994.
- [McM03] K. L. McMillan. Interpolation and SAT-based model checking. In Fifteenth Conference on Computer Aided Verification (CAV'03), pages 1{13. Springer-Verlag, Berlin, July 2003. LNCS 2725.

A new FP6 Network of Excellence: REVERSE - Reasoning on the Web with Rules and Semantics

European News

Luis Moniz Pereira
New University of Lisbon

A new FP6 network of excellence is being established, to start beginning of 2004 (the contract with the Commission is expected to be signed in the course of November or December 2003), whose addressed strategic objectives concern "Semantic-Based Knowledge Systems". This news item is meant to inform the Computational Logic community of REVERSE's overall objectives and participants.

Summary

The objective of REVERSE is to establish Europe as a leader in reasoning languages for the Web by (1) networking and structuring a scientific community that needs it; and by (2) providing tangible technological bases that do not exist today for an industrial software development of advanced Web systems and applications.

The community networked and structured by REVERSE will (1) develop a coherent and complete, yet minimal, collection of inter-operable reasoning languages for advanced Web systems and applications; (2) test these languages on context-adaptive Web systems and Web-based decision support systems selected as test-beds for proof-of-concept purposes; bring the proposed languages to the level of open pre-standards amenable to submissions to standardisation bodies such as the W3C.

REVERSE will develop Education and Training activities targeted at Universities as well as Technology Transfer and Awareness activities targeted at the European industry on reasoning languages for Web systems and applications. Reasoning languages for the Web are an emerging technology that does not exist today. This technology will soon represent an essential breakthrough for Web systems and applications. Thus, REVERSE will promote research on an issue of a considerable economical importance. Doing so,

REVERSE will contribute to the international competitiveness of the European industry in an essential field of today's Information Technologies.

REVERSE will establish itself as the world leading *virtual research centre* on reasoning languages and methods for the Web. REVERSE will ensure that this novel technology is fully exploited and translated into real competitive advantages for the European industry.

Network Objectives

Scientific Objective

The objective of REVERSE is to establish Europe as a leader in the area of reasoning languages for Web systems and applications by (1) networking and structuring a scientific community; and by (2) providing tangible technological bases for an industrial software development of advanced Web systems and applications.

Striving for tangible outcomes, REVERSE will (1) develop a coherent and complete, yet minimal, collection of reasoning languages and prototype processors for these languages for advanced Web systems and applications; (2) test these languages and their prototype processors on context-adaptive Web systems and Web-based decision support systems selected as test-beds for proof-of-concept purposes; (3) bring the proposed languages and their prototype processors to the level of (prototypically implemented and tested) open pre-standards amenable to submissions to standardisation bodies.

Reasoning languages for the Web, although already considered and/or prototypically developed in a restricted manner in some research contexts, are still a technology that does not exist today. Such a technology will represent an essential breakthrough for the current Web and the so-called Semantic Web. REVERSE will promote applied research on this issue of a considerable economical importance.

Scientific context and focus

REVERSE strives for advanced Web systems and applications sometimes referred to as Semantic Web, a term coined in 2001 by Tim Berners-Lee et. al. in the article "**The Semantic Web**" in Scientific American (). This term refers to one of the major current endeavors world wide in Information Technologies. Its goal may be



PROGRAMME ALBAN

Programme of Scholarships for Latin Americans in the European Union

The AlBan Programme aims at the reinforcement of the European Union – Latin America co-operation in the area of Higher Education and covers studies for postgraduates as well as higher training for Latin America professionals/future decision-makers, in institutions or centres in the European Union.

The first grantees of the AlBan Programme will enrol in postgraduate – master and doctorate – or higher specialised training from the academic year 2003/2004. The periods of education and training may range from 6 months to 3 years depending on the level and the education/training programme envisaged.

Participant countries are the 15 Member States of the European Union and the following 18 countries of Latin America: Argentina, Bolivia, Brazil, Chile, Colombia, Costa Rica, Cuba, Ecuador, El Salvador, Guatemala, Honduras, Mexico, Nicaragua, Panama, Paraguay, Peru, Uruguay and Venezuela.

Call for Scholarship applications for the academic year 2004/2005

The deadlines associated with the second Call for Scholarship Applications of the Programme AlBan are:

-Paper submission: **05th January 2004** (date as postmark)

'-on-line' submission (via the Portal AlBan): **22nd January 2004, 24H00 CET**

Candidates are advised to refer to the section 'Documents' in order to access to the text of the Call for Scholarship Applications for the academic year 2004/2005, the Guidelines for Applicants 2004 and the Application Forms 2004/2005.

briefly described as enriching the existing Web with meta-data and data processing (and meta-data processing) so as to provide Web-based systems with advanced (so-called intelligent) capabilities, in particular with context-awareness and decision support, strengthening a person centred, everyday use of the Web.

Reasoning languages are essential to advanced Web systems and applications. The advanced capabilities striven for in most Semantic Web application scenarios primarily call for reasoning (also referred to as logic, deductive, or rule-based) capabilities. Such reasoning capabilities are offered by currently developed Semantic Web languages and/or reasoning systems such as DAML+OIL and OWL 2, BPEL4WS 3, BPML, DAML-S 4, ConsVISor 5, JTP 6, and Triple 7. These languages, however, are developed mostly from functionality centred (e.g. ontology reasoning or access validation) or application centred (e.g. Web service retrieval and composition) perspectives.

Complementing these activities, REVERSE promotes a perspective centred on the reasoning techniques (e.g. forward or backward chaining, tableau-like methods, constraint reasoning, etc.). This REVERSE perspective gives rise to recognise the forms of reasoning needed by Web systems and applications that are inherently different, thus making it possible to provide with a minimal collection of complementary and interoperable reasoning languages for theWeb. Recently, Web circles such as the W3C are becoming conscious of a need for functionality and application independent reasoning languages as generic building-stones of Web and Semantic Web systems and applications. REVERSE aims at fulfilling this need.

Information about participant institutions can be found on the newsletter web site. ❖

SEARCHING FOR RESEARCHERS FROM FUTURE MEMBER STATES

European News

Integrating researchers from the future member states more closely into EU research activities was at the centre of discussions when European Research Commissioner Philippe Busquin met ministers and senior officials from the acceding and candidate countries in Brussels last week. The talks focused particularly on participation in the EU's Sixth Framework Programme for research 2002-2006 - known familiarly as FP6.

The Commission has recently published the response to the first calls for proposals under FP6, and the data available shows that the participation of acceding/candidate countries could be improved. Out of more than 100,000 applications, only 13,000 are from acceding/candidate countries - less than 13%. The corresponding figure for member states is 19%. And only 1,500 organizations from the future member states have been selected for funding, mostly in nano-technology, information society, energy and transport projects. The lowest success rate was in aeronautics and space - just 1.3%.

"Since the very beginning of the Sixth framework programme, acceding and candidate countries have participated in EU research schemes on an equal footing with EU member states", said Commissioner Busquin. "The objective is to ensure the further integration of these countries in the European Research Area. As we can all benefit from the high-level scientific potential these countries have in many areas, we must ensure their participation in the Framework Programme reflects their real potential. There is still scope for improvement, and I am confident research players in EU Member States and in the acceding and candidate countries alike, will make an additional effort to meet this goal."

FP6 is the Union's main instrument for the funding of research in Europe. It encourages closer links between researchers, the pooling of resources, and the bringing together of research teams in different countries. This is seen by the EU as essential if the Union and acceding/candidate countries are to compete, both

scientifically and economically, in the global marketplace.

All acceding and candidate countries are associated to the EU Framework Programme and enjoy the same rights and obligations as the member states. Bulgaria, Czech Republic, Hungary Latvia, Romania, Slovak Republic and Slovenia are also linked to the Euratom Framework Programme. Acceding and candidate countries contribute to, and share, FP6's €20 billion budget with the EU's existing member states.

Anticipating the difficulties that have limited their participation in the Fifth Framework Programme (which ran from 1998 to 2002), a special call for supporting actions was published in April, with a budget of €13 million. These actions are aimed at stimulating, encouraging and facilitating participation in the activities of the priority thematic areas. 201 proposals from all acceding and candidate countries were received before the June closing date. Evaluations took place in September and the selection procedure will be completed shortly. Activities to be supported include organization of conferences and information days, networking of national contact points, setting up of databases, and initiatives to promote the participation of smaller firms. Further measures are also in hand to improve the flow of information on FP6 to the future member states, and a conference on the participation of acceding and countries in FP6 will take place in Bucharest on 12-13 February 2004. ❖

Continued from page 5

CALENDAR OF EVENTS

9TH ESTONIAN WINTER SCHOOL IN COMPUTER SCIENCE, EWSCS'04

PALMSE, ESTONIA

2004, FEB 29 – MAR 5

CALL FOR SUBMISSION OF ABSTRACTS DEADLINE: 16 JANUARY

FQAS 2004 - SIXTH INTERNATIONAL CONFERENCE ON FLEXIBLE QUERY ANSWERING SYSTEMS

LYON, FRANCE

2004, JUNE 24-26

FOR SUBMISSION OF ABSTRACTS DEADLINE: 1 JANUARY

COMBLOG'04

LISBON, PORTUGAL

2004, JULY 28-30

Implementing Rational Features for Agents in Logic Programming

Logic and Multi-Agent Systems

Luis Moniz Pereira
New University of Lisbon

Introduction

We have implemented the following Rational Agent Features: (1) DLP - Dynamic Logic Programming, (2) PDLP – DLP with preferences, (3) MDLP - Multi-Dimensional DLP, (4) LUPS - Language for Dynamic Updates, (5) EVOLP – Evolving Logic Programs, (6) Prolog based standard XML tools. Some of these are further detailed below.

DLP is a semantics for updates of LPs by LP rules. It guarantees that most recent rules are set in force, and previous rules valid by inertia insofar as possible, i.e. are kept for as long as they do not conflict with more recent ones. Originally, in DLP default negation is treated as in the stable models semantics of generalized programs. Now it is also defined for the WFS.

EVOLP is a Logic Programming language for: specifying evolution of knowledge bases; allowing dynamic updates of specifications; capable of dealing with external events; dealing with sequences of sets of EVOLP rules. These rules are generalized LP rules (i.e. possibly with *nots* in heads) plus the special predicate *assert/1* that can appear both in heads or bodies of rules. The argument of *assert/1* can be a full-blown EVOLP rule. The meaning of a sequence of update rules is given by sequences of models. Each sequence determines a possible evolution of the KB. Each model determines what is true after a number of evolution steps (i.e. a state) in the sequence:

- A first model in a sequence is built by “computing” the semantics of the first EVOLP program, where *assert/1* is as any other predicate.
- If *assert (Rule)* is true at some state, then the KB must be updated with *Rule*.
- This updating of the KB, and the “computation” of the

next model in the sequence, is done as in DLP.

An example application concerns a personal assistant agent for email management able to: *Perform basic actions of sending, receiving, deleting messages; Storing and moving messages between folders; Filtering spam messages; Sending automatic replies and forwarding; Notifying the user of special situations.*

All of this dependent on user specified criteria, and where the specification may change dynamically^[4].

We can integrate within the same logic programming framework incomplete, uncertain and paraconsistent reasoning forms. Furthermore, our semantics are able to detect the dependencies on contradiction^[5]. Existing embeddings of other formalisms into our framework are: Ordinary Horn clauses; Generalized Annotated Logic Programs; Fuzzy Logic Programs; Probabilistic Deductive Databases; Weighted Logic Programs and Statistical Defaults; Hybrid Probabilistic Logic Programs; Possibilistic Logic Programs; Quantitative Rules; Multi-adjoint Logic Programming; Rough Sets.

Our XML tools:

- Non-validating XML parser with support for XML Namespaces, XML Base, complying with the recommendations of XML Info Sets. Reads US-ASCII, UTF-8, UTF-16, and ISO-8859-1 encodings.
- Converter of XML to Prolog terms.
- RuleML compiler for the Hornlog fragment, extended with default and explicit negation.
- Query evaluation procedures for Paraconsistent Well-founded Semantics with Explicit Negation.

These and the tools mentioned below, enable our group with possibilities regarding Semantic Web Applications of Logic Programming. This is being pursued in the wider context of the REVERSE NoE submitted to the FP6 (under evaluation but with good chances of approval, and having already passed the first hurdle). Our Logic Programming and the Semantic Web tools include:

- RuleML standards.
- Implementation of Prolog based standard XML tools, namely a fully functional RuleML compiler for the Horn fragment with two types of negation (default and explicit).

SUMMER SCHOOL AND WORKSHOP AT TU DRESDEN

Michael Fischer
University of Liverpool

From June 23 till July 4, 2003, the "Summer School and Workshop on Proof Theory, Computation and Complexity" was organized at the TU Dresden by the members of the research group of Steffen Hölldobler -- Paola Bruscoli, Bertram Fronhöfer, Alessio Guglielmi, Charles Stewart, Sylvia Epp, Mariana Stantcheva, Aning Song -- and with external support by Birgit Elbl from the Universität der Bundeswehr München and Reinhard Kahle from the Universidade Nova de Lisboa.

At the summer school several highly qualified lecturers gave courses: Peter Aczel (Manchester, UK), Roy Dyckhoff (St Andrews, UK), Achim Jung (Birmingham, UK), Sara Negri and Jan von Plato (Helsinki, Finland), Stephen Simpson (Penn State, USA), Jim Lipton (Wesleyan, USA) and Reinhard Kahle (Universidade Nova de Lisboa). The last two days were devoted to a scientific workshop.

The summer school received funding from various sides: the International Quality Network 'Rational mobile agents and systems of agents', the Graduiertenkolleg 334 'Specification of discrete processes and systems of processes by operational models and logics', the Consolato Generale d'Italia - Lipsia/Italienisches Generalkonsulat in Leipzig and also CoLogNet.

However, the most generous funding were 23,400 Euro for participation grants from the German Academic Exchange Service (DAAD) within a newly established funding program. This funding program, projected for many years, aims at developing under the rubric "Deutsche Sommer-Akademie / German Summer-Academy" a well-balanced set of German summer schools which are competitive on the international level. The funding from DAAD shall foster the participation of young foreign researchers at the summer school. At 1450 Euro, the participation grants from DAAD are relatively high and reserved for promising researchers.

After 2001 and 2002 this is the third event in a series of Summer Schools on Computational Logic at TU Dresden which shall be continued in 2004.

- Evolution and updating of knowledge bases. The existing implementations are being integrated with RuleML.

- Semantics of logic programming. Supporting uncertain, incomplete, and paraconsistent reasoning (based on Well-founded Semantics and Answer Sets).

- Development of advanced Prolog compilers (GNU-Prolog and XSB).

- Development of distributed tabled query procedures for RuleML.

- Constraint Logic Programming.

The W4 project

The W4 project aims at developing Standard Prolog inter-operable tools for supporting distributed, secure, and integrated reasoning activities in the Semantic Web. The project goals are:

- Development of Prolog technology for XML, RDF, and RuleML.

- Development of a General Semantic framework for RuleML including default and explicit negation, supporting uncertain, incomplete, and paraconsistent reasoning.

- Development of distributed query evaluation procedures for RuleML, based on tabulation, according to the previous semantics.

- Development of Dynamic Semantics for evolution/update of Rule ML knowledge bases.

- Integration of different semantics in Rule ML (namely, Well-founded Semantics, Answer Sets, Fuzzy Logic Programming, Annotated Logic Programming, and Probabilistic Logic Programming).

-Why have we chosen the Well-founded Semantics with tabling? Because:

-THE adopted semantics for definite, acyclic and (locally) stratified logic programs.

-Defined for every normal logic program, i.e. with default negation in the bodies.

-Polynomial data complexity.

-Efficient existing implementations, namely the SLG-

WAM engine implemented in XSB. Good structural properties.

-It has an undefined truth-value...

-Many extensions exist over WFS, capturing paraconsistent, incomplete and uncertain reasoning.

-Update semantics via Dynamic Logic Programs.

-It can be readily "combined" with DBMSs, Prolog, and Stable Models engines.

-The existence of an *undefined* logical value is fundamental. While waiting for the answers to a remote goal invocation it can be assumed that its truth-value is undefined, and proceed the computation locally. Loops through default negation are dealt with in XSB, via goal suspension and resume operations.

-Tabling IS the right, successful, and available implementation technique to ensure better termination properties and polynomial complexity. Tabling is also a good way to address distributed query evaluation of definite and normal logic programs.

The major guidelines of the project are:

-Tractability of the underlying reasoning machinery.

-Build upon well-understood existing technology and theory, and widely accepted core semantics.

-General enough to accommodate and integrate several major reasoning forms.

-Should extend definite logic programming (Horn clauses). Desirable integration with (logic) functional languages.

-Most of the reasoning should be local (not very deep dependencies among goals at different locations).

-Fully distributed architecture, resorting to accepted standards, recommendations and protocols. Indeed, we have implemented and defined a general and "open" architecture for distributed tabled query-evaluation of definite logic programs. It has a low message complexity overhead. The architecture assumes two types of main components: table storage clients and prover clients. It addresses the issue of table completion by resorting to known termination detection distributed algorithms. It can immediately be extended to handle stratified negation.

The construction of prototypical systems depends on the definition of: Syntactic extensions (apparently, not very difficult); Goal invocation method (Namespaces, XLinks, SOAP, etc.) ; Selection of distributed query evaluation algorithms and corresponding protocols; Formatting of answers and substitutions (should be XML documents); Integration with ontologies. Further applications, testing, and evaluation is required for the construction of practical systems.

Conclusion

In our opinion, Well-founded semantics should be a major player in RuleML, properly integrated with Stable Models. A full-blown theory is available for important extensions of standard WFS/SMs, addressing many of the open issues of the Semantic Web. Most extensions resort to polynomial program transformations, namely those for evolution and update of knowledge bases. They can handle uncertainty, incompleteness, and paraconsistency. Efficient implementation technology exists, and important progress has been made in distributed query evaluation. An open, fully distributed, architecture is being proposed. ❖

Footnotes

1. An updated summary of a presentation at 'Logic-Based Agent Implementation', *An AgentLink/CologNet Symposium*, 3rd February, 2003, Barcelona, España. This work has been developed with contributions by (cf. publications): Jo?o Alcbntara, Jos? J?lio Alferes, Antsnio Brogi, Carlos Damasio, Jo?o Leite, Lu?s Moniz Pereira, Teodor Przymusinski, Halina Przymusinska, Paulo Quaresma.
2. E-mail: imp@di.fct.unl.pt URL: <http://centria.di.fct.unl.pt/~imp>
3. Available at <http://centria.fct.unl.pt/~jja/updates/>
4. Cf. J. J. Alferes, A. Brogi, J. A. Leite, L. M. Pereira, *Logic Programming for Evolving Agents, Cooperative Information Agents (CIA0'3), Helsinki, Finland, August 2003*. And, by the same authors, *An Evolvable Rule-Based E-mail Agent* (submitted).
5. J. Alcbntara, C. V. Damasio, L. M. Pereira, *An Encompassing Framework for Paraconsistent Logic Programs, Journal of Applied Logic, to appear, 2003*. C. V. Damasio, L. M. Pereira, *Hybrid Probabilistic Logic Programs as Residuated Logic Programs, Special issue on Logics in Artificial Intelligence, Studia Logica, 72(2):113-118, 2002*.

International Conference TABLEAUX 2003

Automated Reasoning

Regimantas Pliuskėvicius
Institute of Mathematics and Informatics

The International Conference TABLEAUX 2003 is a continuation of annual international meetings on Automated Reasoning with Analytic Tableaux and Related Methods held since 1992. TABLEAUX 2003 was co-located with the International Conference on Theorem Proving in Higher Order Logics (TPHOLS 2003) and the 11th Symposium on the Integration of Symbolic Computation and Mechanized Reasoning (Calculemus 2003). The three events run in parallel provided the opportunity for contacts with a broader community, corroborated by the joint panel discussion and the talk by Thierry Coquand (Goteborg University, Sweden), jointly invited by Calculemus 2003 and TABLEAUX 2003. The talk "Reasoning about proof search specification" by Dale Miller (INRIA, France), invited by TPHOL, should be mentioned too.

Conference TABLEAUX 2003 brought together researchers and practitioners working on both theoretical and practical aspects of the mechanization of reasoning with tableaux and related methods. Tableaux and related methods such as Gentzen calculi are a convenient and effective formalism for automating deduction not only in classical logic but also in various non-standard logics. Results presented in conference include theoretical foundations, implementation techniques, system development for classical, modal, temporal, intuitionist, non-monotone, conditional, paraconsistent, many-valued, intermediate and description logics. Areas of application of these investigations include verification of software and computer systems, deductive databases, knowledge representation and its required inference engines, and system diagnosis.

The technical program of the conference consisted of 3 invited talks, 14 regular papers, 3 tutorials, 8 position papers and 6 system descriptions. The regular papers and system descriptions presented at the conference were published in the Lecture Notes in Artificial Intelligence (LNAI), vol. 2796, 2003. Position papers



The conferences took place at the main seat of the Faculty of Engineering. On the picture you see the old refectory, recently restored, which is used for the most important ceremonies in the life of the Faculty

and tutorials were published by ARACNE Editrice S.R.L. (Italy), as Technical Report RT-DIA-80-2003, Dipartimento di Informatica e Automazione, Università degli Studi di Roma Tre.

Let us review the invited talks presented at TABLEAUX 2003. J. Schumann (NASA Ames Research Center, USA) presented interesting talk "Automated theorem proving in generation, verification, and certification of safety critical code". In this talk J. Schumann reported on automatic program synthesis systems for state estimation and navigation, AUTOFILTER, and data analysis, AUTOBAYES. These tools automatically generate documented C/C++ code from high-level specifications written in compact domain-specific language. Program synthesis systems are based on some logical inference mechanism, graph reasoning, symbolic algebra and rewriting techniques. For safety-critical applications, the generated code must be certified. It is required the code producer to provide formal proofs that the code satisfies certain safety properties. The system generates some verification conditions which are processed by automatic theorem prover e-SETHEO. M.Abrusci (Rome University, Italy)

presented talk "Non commutative logic: A survey". Non commutative logic (NL) has been introduced by Abrusci and Ruet in 2001. NL is a refinement of Girard's Linear Logic and a conservative extension of Lambek Calculus. NL allows to deal with commutative and non commutative conjunctions and disjunctions. The talk surveys the main results obtained in NL by several authors during 2001-2003, concerning proof-nets, sequent calculus, proof search, completeness theorem and others.

T. Coquand (Goteborg University, Sweden) presented talk "Dynamical method in algebra: A survey". In this talk T.Coquand presented a possible realization of Hilbert's program for some part of abstract algebra. There is a method allowing computations in the algebraic closure of a field, though it is known that such a closure may fail to exist constructively. The talk surveys some results connected with this phenomenon. In the talk the notion of "geometric logic" is presented. The proof in this logic is constructed in some informal way. In some talks presented at the TABLEAUX 2003 it was stressed that it has been developed a number of successful automated deduction systems and methods based on tableau and sequent paradigm. These systems and methods have more rich structure than resolution based systems.

In TABLEAUX 2003 I have presented the position paper (with A. Pliuskeviciene) "Decision Procedure for a Fragment of FTL with Equality". As far as we know, the presented decision procedure is the first one for first-order linear temporal logic with equality. We are grateful the CoLogNET and EU contribution under the IST-FET programme for financial support that allows me to participate in the conference TABLEAUX 2003 and present these results. ❖

VISIT THE NEWSLETTER WEB SITE

For further information on CoLogNET's news visit our web site at www2.cs.ucy.ac.cy/projects/colognet

Logic and Law

Work package 10

Dov Gabbay
Kings College, UK

This article explains the importance of the new emerging area of logic and law and outlines the plans for this work package.

In the past 30 years major evolutionary changes in logic have taken place. Whereas in the first half of the last century logic was mainly applied to mathematics and philosophy, the rise of computer science, artificial intelligence, computational and logical linguistics, logic in engineering, etc, gave logic a big push and accelerated its evolutionary development. All of this is well known and indeed there are many work packages in this project taking care of these areas. What is not covered in the project is the influence and potential significant interaction of these developments on the area of logic and law.

Let us look more closely at the way logic has evolved in response to the needs of computer science, AI and language. These areas have to do with daily human behavior, reasoning and actions. These areas deal with devices and artifacts that help and/or replace the human in his daily activities. Logic is needed partly as the underlying formal language and partly to model and analyze the human in his daily activities to help build better devices to serve, regulate or understand him.

Once logic has evolved in this direction and has developed new logical tools for this purpose, these same kinds of new logics and new tools can help the area of law. Law also deals with humans in their daily activities. Many areas in law require similar additional logical tools as those already available.

Here lies the connection between logic and law. We can say without serious exaggeration that the area of logic and law is going to be central to the further advancement of logic in the next twenty years. If only we can bring the respective communities together and make them aware of their potential! This is why we need this new work package of logic and law now.

It is astonishing to realize that very few people are aware of the true potential of the interaction of the new logics and law. There are many reasons for that, most

ERASMUS MUNDUS: OPENING UP EUROPE'S UNIVERSITIES TO THE WORLD

Parliament adopted a legislative resolution on the setting up a new EU higher education programme - Erasmus Mundus, increasing the budget to €230 million. Whereas the existing Erasmus programme promotes university exchanges within the European Union, Erasmus Mundus seeks to open up Europe's universities and higher education establishments to students throughout the world. The new programme will cover a five-year period from 1 January 2004 to 31 December 2008.

Whereas the "traditional" Erasmus programme supports higher education cooperation within the European Union, Erasmus Mundus seeks to open up Europe's universities and higher education institutions to students from other parts of the world. The programme, scheduled to start in the autumn of 2004.

At first reading in April 2003, Parliament called for the budget of €200 million suggested by the Commission to be increased to €300 million provided this did not affect existing programmes and was within the limits laid down by the financial perspective. However, in its common position in June, the Council put forward a figure of €180 million. Parliament proposes, following lengthy talks with the Council, to set the budget at €230 million. The House hopes that this figure will be accepted by the Council and that the co-decision procedure can thus be concluded at second reading without going to conciliation.

The programme will provide grants for more than 4,000 postgraduate students from non-EU countries other than the EEA/EFTA and accession countries as well as around 1,000 academics. MEPs are keen for the Erasmus Mundus Masters Courses to involve exposure to at least two EU languages. This programme will enable students to travel around Europe attending several different universities. This new European dimension to higher education should be taken into account in the review of existing programmes such as Socrates (Erasmus), in order to take adequate measures to promote access to this programme for European students.

of them social. The new developments in logic are slow to spread around even among logicians. Certainly among researchers in legal reasoning and law theory, many of whom still think of "logic" as "Aristotelian syllogism".

Some bridging work between law and logic was done by C.H. Perelman, who kept in touch with both logicians and judges and lawyers arguing that logic should play a different role. But in his days, the new logical tools were not available as they are now.

The rise of Horn clause logic programming in the 1980s has helped turn some logicians in the direction of the law but early attempts to apply logic to law, such as the formalization of the British Nationality Act, drew a strong critical reaction from the law community on account that Horn clause logic is not rich enough to allow for the wealth of nuances and interpretations/explanation/ revision so common in legal reasoning.

This criticism may have been valid in 1980, it is no longer valid now, especially in view of many advances made in logics of practical reasoning and argumentation. The logic programmers and deontic logicians continued to take interest in law, have their own conferences and journals, but I doubt if they are aware as a community of all relevant developments in logic. They certainly, I think, do not realize (or believe) that law is an area of potentially evolutionary significance to logic.

We therefore need to take steps to promote logic and law and make all relevant communities strongly aware of its potential. Our workpackage proposes certain obvious measures such as:

- An International conference
- A special Journal issue
- An information website

We also propose an unusual step, which will help to consolidate and support the above measures. This step is the preparation of a document/position paper/editorial outlining the research potential and interactive possibilities in the area of logic and law. This document requires effort and vision but is possible to produce because the right people are available in the project. We are also proposing to start a Handbook of Logic and Law, which will serve as a continuing focus for this area. ❖

Formal Requirements Engineering using Observer Models

Andreas Nonnengart, Georg Rock, and Werner Stephan
German Research Centre for Artificial Intelligence

Abstract

Today we are confronted with an enormous variety of formal software engineering approaches and tools. Among these are many that address the critical early stages of software development. However, only little attention has been paid to the integration of different specialized approaches and to the overall development process. In this article we introduce a technique for formal requirements analysis (observer models) that deals with particular perspectives on a system rather than with particular aspects of it.

Introduction

Thirty years ago (in the early seventies of the last century) the question what are formal methods was easy to answer. However, code verification in Hoare style systems failed not only because of its inability to cope with complexity but also because this restricted approach did not meet important needs of software engineering. Meanwhile the situation has drastically changed. We are confronted with an enormous variety of formal approaches and tools. Among these are many that address the critical early stages of software development. Although much progress has been made with respect to fully automatic methods little attention has been paid to the integration of different specialised approaches and to the overall development process. Formal techniques for requirements analysis often deal with a particular aspect of the system to be designed. Examples of such aspects are properties of information flow, correctness of (cryptographic) protocols, and real-time behaviour. We concentrate on real-time analysis as one view on a system among others. It is not difficult to imagine a system that separates different applications by controlling the flow of information between them using authentication protocols as one of the security mechanisms and that in addition has to satisfy certain real-time requirements.

Although it is well known that many severe errors occur

already in the early stages of the development process, also later design stages like architectural design and implementation have to be treated formally if one aims at high assurance levels. For example, according to level EAL5 of Common Criteria (CC) [7], a formal high-level design and a “correspondence proof” with respect to the so-called functional specification has to be provided. In the case of real-time systems one has to define how the intended global behaviour is realized by separating the control component from its environment and by making assumptions (delays, cycle time) explicit. Therefore, a number of particular views for requirements analysis have to be linked to a single abstract system specification (System Design Spec.) that serves as a starting point for the refinement process (see Figure 1).

Rather than having a satisfies relation between a specification and a collection of simple properties that have to be established, requirements analysis will be based on its own descriptions (views) and postulated properties that refer to these descriptions. The description of a particular view will not necessarily use the same terminology as the system specification and often application specific formalisms and tools will allow for an efficient analysis. For establishing information flow properties a technique called non-interference analysis, which is based on closure properties of sets of (system) traces, has to be used [8]. The analysis of protocols is based on a different kind of traces [10] that include steps of an attacker. A number of tools, like for example the one described in [9], have been used in this area. The real-time view can be implemented by Hybrid Automata [2] or by Timed Automata [3]. Tools like HyTech [4] provide efficient techniques to establish real-time properties.

We introduce a general technique called observer models to link abstract descriptions of the real-time behaviour of a system to a system specification that consists of a control component, an environment, and a clock by means of an observer mapping. We outline the general technique and refer the interested user for more details to [11, 12].

Observer Models for Real-Time Properties

Requirements of a system to be developed are specified and analysed by possibly several different

formalisms that are specific for a particular view on that system. The choice of the formalisms can be influenced by several factors:

- Preferences or previous knowledge of the user,
- special requirements, which presupposes a certain power of the specification language,
- available system support for this language or
- re-use of requirements specifications.

In Figure 1 each view is represented by an Observer-Spec_i following its own description technique and formalism. One of these specifications might contain a global description of the runs of a protocol while another view concentrates on real-time properties. In the following we shall assume that the real-time view is given by Hybrid Automata [2].

As already mentioned above a view will also include properties, called OS-Properties_i in Figure 1, that have to be established from the observer specification.

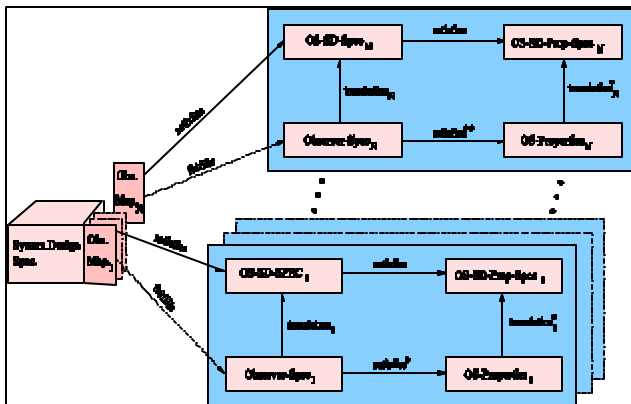


Figure 1: Observer models

For example, real-time requirements can be formulated and proven using tools like HyTech [4]. Note that we consider Hybrid Automata as a kind of comprehensive description of the entire system behaviour with respect to time constraints. As can be seen from our example, the description is global in the sense that it does not distinguish between the control system and its environment. States of the Hybrid Automaton therefore do not directly correspond to internal states of the System Design Spec. They rather describe certain situations that might occur in a run of the components (controller, environment, clock) together. To demonstrate this we will start an example scenario in Section 3.

To integrate various views into a common formal

development the Observer-Spec and the OS-Properties_i first have to be translated into the language of the System Design Spec. The resulting specifications are called OS-SD-Spec and OS-SD-Prop-Spec. The translation of Hybrid Automata into the specification language of VSE-II is described in [11, 12]. This integration of Hybrid Automata into VSE-II results in a combination of interactive and automatic verification techniques whereas it is possible to use the automatic verification results in the interactive approach and vice versa. The embedding is achieved by an exact discretisation of dense real-time behaviours of Hybrid Automata such that VSE-II can cope with them. This discretisation is defined such that it is not just an approximation but rather mirrors dense behaviour exactly and that without an explicit introduction of rational numbers. The language used in VSE-II [6, 5] is VSE-SL. It is similar to TLA but has some extensions concerning shared variables and assumption commitment specifications, among others.

It can be shown that the satisfies relation between OS-SD-SPEC_i and OS-SD-Prop-Spec_i holds, if and only if the satisfies relation holds between Observer-Spec_i and OS-Properties_i. First of all this means that results obtained by using a tool like HyTech can be safely integrated into the overall development. However, since the language of OS-SD-Spec and OS-SD-Prop-Spec is more expressive (than that of Hybrid Automata) requirements specifications that are still “in the style of Hybrid Automata” but more general than these can be used in this context if one is inclined to use deductive techniques instead of model-checking. As an example one might wish to consider state transitions where one of the parameters (like speed) changes arbitrarily.

We are still left with the problem of establishing a link between the system specification and the particular real-time view we have defined. This is done by a mapping (called Obs. Map_i in Figure 1) that (in the case of real-time requirements) interprets a given state of the interleaved computation of the controller, environment, and clock as a state of the (translation of the) Hybrid Automaton. It thereby turns the entire System Design Spec. scenario into a model (in the sense of Hybrid Automata). For this we need to be sure that the translation faithfully preserves the

semantics of Hybrid Automata.

In the following we outline an instantiation of the general methodology that uses an abstract and global specification of the real-time behaviour of a gasburner scenario by a Hybrid Automaton on the one side and the specification of a controller that is connected to an environment by sensors and actors on the other.

General Specification Scheme for Observer Models

The general scenario (see Figure 2) consists of three components: an environment, a controller and an observer/clock component. Generally, given a system design one cannot always accurately decide, which parts are to be assigned to the environment and which parts belong to the controller. In the application of formal methods we are often interested in the safety critical parts of the system to be developed. The other parts are considered to be irrelevant for the safety of the system. These parts could consist for example of monitoring units. The fact is substantial that the control, which is to be refined later or at least is applicable for a refinement, should contain all the safety critical parts.

The behaviour of the environment is determined by the specification of its interfaces. I.e. the environment has to supply the values needed on the various interfaces (in time). To guarantee the right functioning of the system we have to make assumptions about the correct

range, and depth of the specification of both the environment and controller depend on the properties that should be fulfilled.

The environment and the system/controller are specified as temporal logic specifications. Both components can be structured into subcomponents. Their communication is indicated by arrows in Figure 2. The specification of the observer/clock component differs in some sense from these usual specifications. One of the tasks of the observer is that it holds the time. But this fact does not influence the method described here. The essential part of the observer is that it observes the system parts that are built from the Controller and the Environment components. These observations are filtered by the observer and communicated to the outside world. This filtration of the behaviour of the whole system constitutes a special view on the system that is a real-time view represented by an Hybrid Automaton specification. This is indicated by the right part of Figure 2 consisting of ObserverSpec 1 (instantiated by the Hybrid Automaton Gasburner (see Figure 4)) and the translation of this gasburner into a VSE-II specification (see OS-SD-SPEC 1 in Figure 2). The languages used in the real specification are VSE-SL (VSE-Specification Language) and Hybrid Automata as indicated in Figure 2.

The General Scenario in VSE-II

The implementation of the general scenario in VSE-II is illustrated in Figure 3.

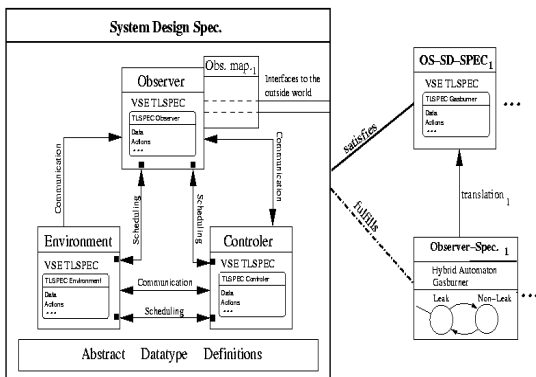


Figure 2: General Scenario

behaviour of the environment. These assumptions can be used in the proof of the postulated properties of the system. If the environment component does not only exist as an interface definition, but also as a component with accurately specified behaviour, then one can prove these assumptions about the environment using the behaviour of the environment. Of course the type,

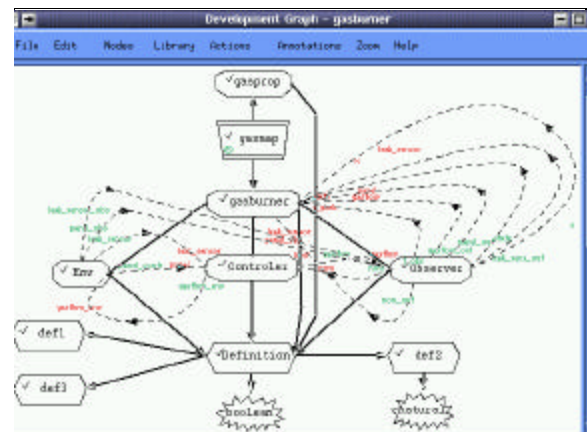


Fig. 3: Development Graph of the Real Gasburner

It represents a real gasburner specification that consists of the three components Env, Controller and Observer that are implemented as VSE-II Temporal the Logic Specifications. These components are composed

to the gasburner system. The safety model of this specification is represented by the temporal logic gasprop that results from the translation of well-known Hybrid Automaton gasburner (see Figure 4) to VSE-II. Thus the Hybrid Automaton gasburner specification represents a view on a real gasburner specification whereas such a real gasburner in general does not talk about states like leaking or non leaking. It is the responsibility of the observer to map the states of the real gasburner to the states of the Hybrid Automaton gasburner.

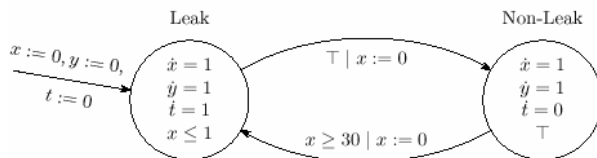


Fig. 4: Gasburner as Hybrid Automaton

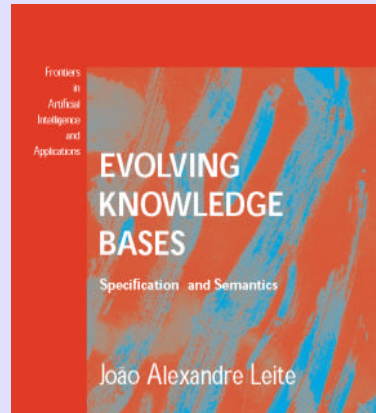
Conclusion and Future Work

We have sketched a methodology, observer models, for formal requirements engineering. Its applicability is illustrated with the help of a realistic gasburner example. One of the open issues in this context is the question how to refine a specification without re-doing the whole proof work. This problem seems to be very similar to that of refinement in the security area, for example in protocol analysis.

References

1. Mart'ın Abadi and Leslie Lamport. The Existence of Refinement Mappings. *Theoretical Computer Science*, 82(2):253–284, May 1991.
2. R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138:3–34, 1995.
3. R. Alur and D. L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.
4. T. A. Henzinger and P.-H. Ho. HyTech : The cornell hybrid technology tool. In P. Antsaklis, A. Nerode, W. Kohn, and S. Sastry, editors, *Hybrid Systems II*, pages 265–293. Springer Verlag, Lecture Notes in Computer Science, vol. 999, 1995.
5. D. Hutter, B. Langenstein, J. H. Siekmann C. Sengler, W. Stephan, and A. Wolpers. Deduction in the verification support environment (vse). In *Formal Methods Europe (FME)*, LNCS. Springer, 1996.
6. Dieter Hutter, Heiko Mantel, Georg Rock, Werner Stephan, Andreas Wolpers, Michael Balsler, Wolfgang Reif, Gerhard Schellhorn, and Kurt Stenzel. VSE: Controlling the complexity in formal software developments. In D. Hutter, W. Stephan, P. Traverso, and M. Ullmann, editors, *Proceedings Current Trends in Applied Formal Methods, FM-Trends 98*, Boppard, Germany, 1999. Springer-Verlag, LNCS1641.
7. U. Institute and o Standards. Common criteria for information technology security evaluation, 1999.
8. Heiko Mantel. Possibilistic definitions of security - an assembly kit. In *Proceedings of the 13th IEEE Computer Security Foundations Workshop*, Cambridge, England, 2000. IEEE Computer Society Press.
9. Catherine Meadows. The NRL protocol analyzer: An overview. *Journal of Logic Programming*, 26(2):113–131, 1996.
10. Jonathan K. Millen. CAPSL: Common authentication protocol specification language. The MITRE Corporation, Technical Report MP 97B48, 1997. <http://www.csl.sri.com/millen/capsl>.
11. A. Nonnengart, G. Rock, and W. Stephan. Expressing Realtime Properties in VSE-II. In *ESA Workshop on On-Board Autonomy*, volume WPP-191, pages 447–454, October 2001.
12. A. Nonnengart, G. Rock, and W. Stephan. Using Hybrid Automata to Express Realtime Properties in VSE-II. In Ingrid Russel and John Kolen, editors, *Proceedings of the Fourteenth International Florida Artificial Intelligence Research Society Conference*, pages 640–644. AAAI Press, 2001. ❖

A BOOK REVIEW



BOOK TITLE: Evolving Knowledge Bases - Specification and Semantics,

Volume 81 Frontiers in Artificial Intelligence and Applications

AUTHOR

João Alexandre Leite

BOOK INFO

This book consists of the author's PhD Thesis ;

ABOUT THE AUTHOR

João Alexandre Leite is Assistant Professor at the Department of Computer Science of the New University of Lisbon, in Portugal, and member of CENTRIA, its Artificial Intelligence Centre.

BOOK DESCRIPTION

In this book, the author incrementally specifies, semantically characterizes, and illustrates with examples, the concepts and tools necessary to the development of Evolving Knowledge Bases (EKB). An EKB is a knowledge base which can not only be externally updated, but is also capable of self evolution by means of its internally specified behaviour. To this purpose, the author first defines the notion of Dynamic Logic Programming, based on the concept of Logic Program Updates, which characterizes knowledge given by a sequence of logic programs, each representing a state of the world. Then, he sets forth a language capable of uniformly specify the external updates as well as the knowledge base's internal behaviour and its updates.

RELATED LINKS

Author's homepage: <http://centria.di.fct.unl.pt>

Department of Computer Science: <http://www.di.fct.unl.pt/>

New University of Lisbon: <http://www.unl.pt/>