Software Model Checking at Design and Implementation *

Fang Yu

Advisor: Tevfik Bultan
Department of Computer Science
University of California, Santa Barbara

November 9, 2006

Abstract

This work is aimed to investigate main techniques of the cutting-edge model/program/design checkers, such that scalable software model checking may be achieved in future research.

1 Introduction

Software model checking is one of the essential techniques to guarantee system correctness. Being a witness of a shift from the verification of abstract hand-built models of codes, towards the direct verification of implementation level code, in this survey we present the main techniques in model checking, as well as a broad survey of modern tools. We aim to achieve scalable software model checking in future research based on this study.

We propose three survey directions: a) model checking, b)program verification, and c)software modeling. We first go through main milestones of model checking in past twenty years, which include symbolic model checking [BCL90, BCMDH90, GP00], abstraction [CGL92], symmetry reduction [ES94, ES97], partial order reduction [CGMP99, KLMPY98], bounded model checking [AKMM03, BCCFZ99, CBRZ01, CGRPST02], induction [MRS03, SSS00], interpolation [McMillan03], predicate abstraction [HJMK04, HJMS02] and refinement [BCDR04]. Methods successfully addressed these issues did expand the ability of model checking and had been widely adopted in model checkers, such as SPIN [Holzmann97], NuSMV [CCGRPST02], and ALV [YBB05], as well as modern program checkers.

*This is the reading list for Fang's Major Area Exam.

While model checkers verify general properties in specified languages, program checkers usually verify specific properties of general languages. At the second part, we broadly survey program checkers: JPF [BHPV00], Bandera [CD00], MOPS [CD02, CDW04], Bebop [BR01], Blast [BMMR01, HJM04, HJMQ03], CMC [MPCED02], CBMC [CKY03, CKL04], VeriSoft [Godefroid97], eXplode [YTEM04,YSE06], and WebSSARI [HYHTLK04]. These checkers differ in their algorithms, target language and verified properties. We will focus on the core algorithms and pros/cons of these checkers.

We then delve into software modeling to seek an efficient framework to facilitate scalable model checking. We first look at general modeling languages: JML[LBR06, BCCEKL-RLP05], UML[OMG99]/OCL[OCL99, WK98], IOA[GL98], STATEMATE[Harel90], MSC[MSC96, HT03]. These modeling languages are used to formally specify system requirements before code generation/implementation. We believe that ensuring that designs are robust and free from conceptual flaws forms a solid foundation of software system development. Previous researches addressing the correctness of design languages include a) Alur's work[AY99] for MSC, b) Bogar[DHHRRW06, RDH06, RRD04, RRDH04] for JML, c) Aloca for Alloy[DCJ06, JSS01, Jackson02, Jackson06] d) Model checking for UML state machines[LM99, SK01] and USE[GBR03] for UML/OCL.

For future research, we aim to take advantage on design language to achieve scalable code-level checking.

2 Model Checking

Reading List:

AKMM03 N. Amla, R. Kurshan, K. McMillan and R. K. Medel, Experimental Analysis of Different Techniques for Bounded Model Checking. In Proc. of TACAS'03, LNCS, Warsaw, Poland, 2003.

BCL90 J. R. Burch, E. M. Clarke, and D. E. Long. Symbolic model checking with partitioned transition relations. In VLSI 91, Edinburgh, Scotland, 1990.

BCMDH90 J.R. Burch, E.M. Clarke, K.L. McMillan, D.L.Dill, L.J. Hwang. Symbolic Model Checking: 10^{20} States and Beyond, IEEE LICS, 1990.

BCDR04 Thomas Ball, Byron Cook, Satyaki Das, Sriram K. Rajamani. Refining Approximations in Software Predicate Abstraction, TACAS 2004.

CBRZ01 E. Clarke, A. Biere, R. Raimi,and Y. Zhu, "Bounded Model Checking Using Satisfiability Solving." In Formal Methods in System Design, July 2001.

CGL92 E. M. Clarke, O. Grumberg, and D. E. Long. Model checking and abstraction. In Proceedings of the Nineteenth Annual ACM Symposium on Principles of Programming Languages, January 1992.

CGMP99 E. Clarke, O. Grumberg, M. Minea, D. Peled. State-Space Reduction using Partial-Ordering Techniques, STTT 2(3), 1999, pp.279-287.

GP00 E. Clarke, O. Grumberg, and D. Peled, Model Checking, MIT Press, Jan. 2000.

ES94 E. A. Emerson and A. P. Sistla, Symmetry and model checking, In Proc. of the International Conference on Computer Aided Verification (CAV93), LNCS 697, pp. 463-478, Elounda, Greece, 1993.

Holzmann97 G. J. Holzmann, The model Checker SPIN, IEEE Transaction on Software Engineering, Vol 23, No. 5, May, 1997, pp. 279-295.

HJMK04 Thomas A. Henzinger, Ranjit Jhala, Rupak Majumdar, and Kenneth L. McMillan. Abstractions from Proofs. Proceedings of the 29th Annual Symposium on Principles of Programming Languages, ACM Press, pp. 232-244, 2004.

KLMPY98 R. P. Kurshan, V. Levin, M. Minea, D. Peled and H. Yenigun, Static Partial Order Reduction, In Proc. of Tools and Algorithms for Construction and Analysis of Syatems (TACAS'98), LNCS 1384, pp. 345-357, Lisbon, 1998.

McMillan03 K. L. McMillan, Interpolation and SAT-Based Model Checking. In Proc. of Computer Aided Verification (CAV'03), LNCS 2725, pp. 1-13, 2003.

MRS03 L. de Moura, H. Rueß and M. Sorea, Bounded Model Checking and Induction: from Refutation to Verification. In Proc. of CAV'03, LNCS 2725, pp. 14-26, 2003.

SSS00 M. Sheeran, S. Singh and G. Stålmarck, Checking Safety Properties Using Induction and a SAT-solver. In Proc. of FMCAD 2000. LNCS 1954:108, 2000.

YBB05 Tuba Yavuz-Kahveci, Constantinos Bartzis, and Tevfik Bultan. Action Language Verifier, Extended. In Proc. of the 17th International Conference on Computer Aided Verification (CAV05), LNCS 3576, pp. 413-427, 2005.

3 Program Verification

Reading List:

BB04 Aysu Betin-Can, Tevfik Bultan: Verifiable Concurrent Programming Using Concurrency Controllers. ASE 2004, pp. 248-257.

BMMR01 T. Ball, R. Majumdar, T. Millstein, S.K. Rajamani, Automatic Predicate Abstraction of C programs, Proc. PLDI 2001.

BR00 Thomas Ball, Sriram K. Rajamani, Bebop: A Symbolic Model Checker for Boolean Programs, SPIN 2000 Workshop on Model Checking of Software, LNCS 1885, August/September 2000, pp. 113-130.

CD00 J.C. Cobett, M.B. Dwyer, et al. Bandera:Extracting finite state models from Java source code, Proc. 22nd Int. Conf . on Software Engineering(ICSE00), pp.439-448.

CD02 Hao Chen and David Wagner. MOPS: an Infrastructure for Examining Security Properties of Software. Proc. ACM Computer and Communications Security 2002.

CDW04 Hao Chen, Drew Dean, and David Wagner, Model Checking One Million Lines of C Code. Network and Distributed System Security (NDSS 2004), February 2004.

CGPL06 Cristian Cadar, Vijay Ganesh, Peter M. Pawlowski, David L. Dill, Dawson R. Engler. EXE: Automatically Generating Inputs of Death, In Proc. of the 13th ACM Conference on Computer

CKY03 E. Clarke, D. Kroening, K. Yorav, "Behavioral Consistency of C and Verilog Programs using Bounded Model Checking." In Proc. of the 40th Design Automation Conference, Session 23.3, Anaheim, CA, 2003.

CKL04 E. Clarke, D. Kroening, and F. Lerda, A Tool for Checking ANSI-C Programs, Proc. Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2004), LNCS 2988, pp. 168-176.

CSD02 Chandrasekhar Boyapati, Sarfraz Khurshid and Darko Marinov. Korat: Automated Testing Based on Java Predicates. ACM/SIGSOFT International Symposium on Software Testing and Analysis (ISSTA), Rome, Italy. July 2002.

Godefroid97 P. Godefroid, VeriSoft: A Tool for the Automatic Analysis of Concurrent Reactive Software. Proc. of the 9th Conference on Computer Aided Verification, LNCS 1254, pp. 476-479, June 1997.

HJM04 Thomas A. Henzinger, Ranjit Jhala, and Rupak Majumdar. Race checking by context inference. In Proceedings of the International Conference on Programming Language Design and Implementation (PLDI), ACM Press, 2004.

HJMS02 Thomas A. Henzinger, Ranjit Jhala, Rupak Majumdar, and Gregoire Sutre. Lazy Abstraction. Proceedings of the 29th Annual Symposium on Principles of Programming Languages, ACM Press, pp. 58-70, 2002.

HJMQ03 Thomas A. Henzinger, Ranjit Jhala, Rupak Majumdar, and Shaz Qadeer. Thread-modular Abstraction Refinement. In Proceedings of the 15th International Conference on Computer-Aided Verification (CAV), LNCS 2725, Springer-Verlag, pages 262-274, 2003.

HYHTLK04a Yao-Wen Huang, Fang Yu, Christian Hang, Chung-Hung Tsai, Der-Tsai Lee, Sy-Yen Kuo. Securing Web Application Code by Static Analysis and Runtime Protection. In: Proceedings of the 13th International World Wide Web Conference (WWW2004), pages 40-52, New York, May 17-22, 2004.

HYHTLK04b Y.-W. Huang, F. Yu, C. Hang, C.-H. Tsai, D.-T. Lee and S.-Y. Kuo, Verifying Web Applications Using Bounded Model Checking. In Proc. of the 2004 International Conference on Dependable Systems and Networks, pages 199-208, Florence, Italy, Jun 28-Jul 1, 2004.

MPCED02 Madanlal Musuvathi, David Y.W. Park, Andy Chou, Dawson R. Engler, David L. Dill. CMC: A pragmatic approach to model checking real code. Proc. Operating System Design and Implementation (OSDI) 2002.

VHBPL03 W. Visser, K. Havelund, G. Brat, S. Park and F. Lerda, Model Checking Programs, Automated Software Engineering Journal Volume 10, Number 2, April 2003

YTEM04 Junfeng Yang, Paul Twohey, Dawson Engler, and Madanlal Musuvathi, Using Model Checking to Find Serious File System Errors, Proc. Operating System Design and Implementation (OSDI) 2004. and Communications Security, 2006.

YSE06 Junfeng Yang, Can Sar, and Dawson Engler, eXplode: a Lightweight, General System for Finding Serious Storage System Errors, Proc. of the 7th Symposium on Operating System Design and Implementation(OSDI), 2006.

4 Software Modeling and Analysis:

Reading List:

AY99 R. Alur and M. Yannakakis. Model checking of message sequence charts. In CONCUR'99: Concurrency Theory, Tenth International Conference, LNCS 1664. pp. 114-129.

BBKRT04 Michael Balser, Simon Baumler, Alexander Knapp, Wolfgang Reif, Andreas Thums. Interactive Verification of UML State Machines. ICFEM 2004. pp. 434-448

BCCEKLRLP05 Lilian Burdy, Yoonsik Cheon, David Cok, Michael Ernst, Joe Kiniry, Gary T. Leavens, K. Rustan, M. Leino, and Erik Poll. An overview of JML tools and applications. International Journal on Software Tools for Technology Transfer, volume 7, number 3, pages 212-232, June 2005.

GBR03 Martin Gogolla, JoLrn Bohling, and Mark Richters. Validation of UML and OCL Models by Automatic Snapshot Generation. In Proc. 6th Int. Conf. Unified Modeling Language (UML'2003). Springer, Berlin, LNCS 2863, 2003.

HLNPPSST90 D. Harel, H. Lachover, A. Naamad, A. Pnueli, M. Politi, R. Sherman, A. Shtul-Trauring and M. Trakhtenbrot, STATEMATE: A Working Environment for the Development of Complex Reactive Systems, IEEE Trans. on Software Engineering 16:4 (1990), pp. 403-414.

Jackson99 Daniel Jackson, A Comparison of Object Modelling Notations: Alloy, UML and Z. MIT Lab for Computer Science. August 11,1999.

Jackson00 Daniel Jackson. Automating First-Order Relational Logic. Proc. ACM SIGSOFT Conf. Foundations of Software Engineering, November 2000.

Jackson02 Daniel Jackson, Alloy: A Lightweight Object Modeling Notation. Proc. in the ACM Transactions on Software Engineering and Methodology, Vol. 11, Issue 2, pages 256-290; April 2002.

Jackson06 Daniel Jackson. Dependable Software by Design. Scientific American. June 2006.

JSS00 Daniel Jackson, Ian Schechter, and Ilya Shlyakhter. Alcoa: the alloy constraint analyzer. In Proc. 22nd International Conference on Software Engineering, Limerick, June 2000.

JSS01 D. Jackson, I. Shlyakhter and M. Sridharan, A Micromodularity Mechanism. Proc. of the Joint 8th European Software Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering. ACM Press, 2001.

LBR99 Gary T. Leavens, Albert L. Baker, and Clyde Ruby. JML: A Notation for Detailed Design. In Haim Kilov, Bernhard Rumpe, and Ian Simmonds (editors), Behavioral Specifications of Businesses and Systems, chapter 12, pages 175-188. Copyright Kluwer, 1999.

LMM99 D. Latella, I. Majzik, and M. Massink. Automatic verification of UML statechart diagrams using the SPIN modelchecker. Formal Aspects of Computing, 11(6):637–664, 1999.

LP99 J. Lilius and I. P. Paltor. Vuml: a tool for verifying uml models. Technical report, Abo Akademi University, 1999.

MCOGKF03 E. Mota, E. Clarke, W. Oliveira, A. Groce, J. Kanda, and M. Falcao. VeriAgent: an Approach to Integrating UML and Formal Verification Tools." In Sixth Brazilian Workshop on Formal Methods (WMF 2003), pp. 111–129, Brazil, October 2003.

OCL99 OMG. Object Constraint Langugae Specificaiton. In OMG Unified Modeling Language Specification, Version 1.3, June 1999.

OMG99 OMG, editor. OMG Unified Modeling Language Specification, Version 1.3, June 1999. Object Management Group, Inc., Framingham, Mass.

RDH06 Robby, Matthew B. Dwyer, John Hatcliff. Bogor: A Flexible Framework for Creating Software Model Checkers, In the Proceedings of Testing: Academic and Industrial Conference - Practice And Research Techniques, June 2006.

RRD04 Robby, Edwin Rodriguez, Matthew B. Dwyer, John Hatcliff. Checking Strong Specifications Using An Extensible Software Model Checking Framework, In the Proceedings of the Tenth International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2004).

RRDH04 Robby, Edwin Rodriguez, Matthew B. Dwyer, John Hatcliff. Checking JML Specifications Using An Extensible Software Model Checking Framework, August 2004. In the International Journal of Software Tools for Technology Transfer (STTT).

MSC96 Z.120. ITU-TS recommendation Z.120: Message Sequence Chart (MSC), 1996.

HT03 D. Harel and P. S. Thiagarajan, Message Sequence Chart. Book Chapter in UML for Real: Design of Embedded Real-time Systems, Kluwer Academic Publishers, 2003.

5 Conclusion and Future Research