Multi-University Research Initiative on
High-Confidence Design for Distributed Embedded Systems

Frameworks and Tools for High-Confidence Design of Adaptive, Distributed Embedded Control Systems

Year 2 Progress Report

Vanderbilt University
Institute for Software Integrated Systems
2015 Terrace Place, Nashville, TN 37203
(615) 322-3455 (office)
(615) 343-7440 (fax)
janos.sztipanovits@vanderbilt.edu

TEAM MEMBERS:

Vanderbilt: J. Sztipanovits (PI) and G. Karsai
UC Berkeley: C. Tomlin (Lead and co-PI), Edward Lee, George Necula and S. Sastry
CMU: Bruce Krogh (Lead and co-PI) and Edmund Clarke
Stanford: Stephen Boyd

FA9550-06-0312

1. Objectives

This project aims to develop a comprehensive approach to the model-based design of high-confidence distributed embedded systems. We will take advantage and fully leverage a shared theoretical foundation and technology infrastructure in four focus areas: hybrid and embedded systems theory, model-based software design, composable tool architectures and experimental testbeds. The objectives of our research in the focus areas are the following:

1. Develop theory of deep composition of hybrid systems with attributes of computational and communication platforms. We will address compositionality, concurrency, heterogeneity and resource, robustness, approximate verification and adaptive control architectures for uncertainty handling.
2. Develop foundations of model-based software design for high-confidence, networked embedded systems applications. We will investigate new semantic foundations for modeling languages and model transformations, precisely architected software and systems platforms that guarantee system properties via construction, and new methods for static source code verification and testing, as well as for dynamic runtime verification and testing.
3. Develop composable tool architecture that supports high-level reusability of modeling, model-analysis, verification and testing tools in domain-specific tool chains. We create new foundation for tool integration that goes beyond data modeling and data transfer.
4. Demonstrate the overall effort by creating an end-to-end design tool chain prototype for the model-based generation and verification of embedded controller code for experimental platforms.

2. Status of the Effort

We have achieved three major breakthroughs in achieving compositionality of control systems on computational and communication platforms. First, we have shown how the linear matrix inequality (LMI) methods can be used to synthesize a constant state-feedback controller that minimizes the performance bound, for a given level of timing jitter. Second, we have extended this method to establish an upper bound on the worst-case performance degradation for networked controllers due to the network delays as a function of the delay bound, which can be used as a design parameter for the networked implementation. Third, we have developed a new theory for networked controller design using the principle of passivity that makes network controllers robust against time variant delays. We have completed the working prototype of an end-to-end tool chain for the model-based design of networked control systems. The underlying implementation platform is the Time-Triggered Architecture (TTA). We have built demonstrations for auto-generating code from verified models. We have completed a Ptolemy code generator using the PTIDES (Programming Temporally Integrated Distributed Embedded System) model. We have achieved significant progress in our STARMAC fully autonomous aircraft by integrating and testing coordinated search control algorithms in the design.

3. Accomplishments and New Findings

We continued our work on developing tools, methods and other components of the project along the four objectives.

3.1 Hybrid and Embedded Systems Theory

3.1.1 Embedded Systems Modeling and Deep Compositionality (Krogh, Tomlin, Sastry)

During the past year, we have continued developing abstraction techniques for real-time systems. The main advantage of this technique is that it reduces the complexity of checking properties of certain types of real-time systems. This enabled applying verification techniques to larger software systems, like the real-time code needed to control UAVs.

3.1.2 Hierarchies of Robust Hybrid and Embedded Systems (Tomlin, Krogh, Sastry)

Reachability analysis. We have extended our technology for reachable-set-based analysis and design of collision avoidance schemes for multiple autonomous quadrotor aircraft, and to the very close formation flying of multiple fixed wing UAVs.

Counterexample-guided abstraction refinement for control parameter design. We developed a new approach to finding the set of parameters for which a given linear hybrid automaton does not reach a given set of bad states. The problem is known to be semi-solvable (if the algorithm terminates the result is correct) by introducing the parameters as state variables and computing the set of reachable states. This is usually too expensive, however, and in our experiments only possible for very simple systems with few parameters. We propose an adaptation of counterexample-guided abstraction refinement (CEGAR) with which one can obtain an underapproximation of the set of good parameters using linear programming. The adaptation is generic and can be applied on top of any CEGAR method where the counterexamples correspond to paths in the concrete system. For each counterexample, the cost incurred by underapproximating the parameters is polynomial in the number of variables, parameters, and the length of the counterexample. We identify a syntactic condition for which the approach is complete in the sense that the underapproximation is empty only if the problem has no solution. Experimental results for two CEGAR methods, a simple discrete version and iterative relaxation abstraction (IRA), both show a drastic improvements in performance compared to standard reachability techniuques.

Extending CEGAR using Craig interpolants. The use of Craig interpolants has enabled the development of powerful hardware and software model checking techniques. Efficient algorithms are known for computing interpolants in rational and real linear arithmetic. We have developed polynomial time algorithms for obtaining interpolants for conjunctions of linear diophantine equations, linear modular equations, and linear diophantine disequations. We have shown the utility of the proposed interpolation algorithms for discovering modular/divisibility predicates in a CEGAR framework. This has enabled verification of simple programs that cannot be checked using existing CEGAR based model checkers.

3.1.3 Verification and Validation of Conservative Approximations (Clarke, Krogh)

Bounded-time verification technique that combines software model checking and simulation. We have developed a technique for verifying safety properties of a system composed of a supervisory controller, implemented with software, interacting with a continuous-time plant. A combination of software model checking and numerical simulation is used to compute a conservative approximation of the reachable states. The technique verifies system properties in the presence of nondeterministic behavior in the software due to, for instance, interleaving of controller tasks. A notion of program equivalence is used to characterize the behaviors of the controller, and the bisimulation functions of Girard and Pappas are employed to characterize the behaviors of the plant. The approach can conservatively merge traces that reach states that are in proximity to each other. The technique has been implemented for the case of affine and polynomial plant dynamics, which allows efficient operations on ellipsoidal sets based on convex optimization involving linear matrix inequalities (LMIs).

Systematic search for counterexamples using model checking and numerical simulation. Model checkers for program verification have enjoyed considerable success in recent years. In the control systems domain, however, they suffer from an inability to account for the physical environment. For control systems, simulation is the most widely used approach for validating system designs. We have developed a new technique for finding counterexamples that uses a software model checker to perform a systematic simulation of the software implementation of a controller coupled with a continuous plant. Instead of performing a large set of independent simulations, our approach uses the model checking notion of state-space exploration by piecing together numerical simulations of the plant and transitions of the controller. Our implementation of this technique uses an explicit-state source-code model checker to analyze the software and the MATLAB/Simulink environment to model and simulate the plant.

3.1.4 Adaptive Control Architectures for Uncertainty Handling (Boyd, Krogh)

Analysis and Synthesis of state-feedback controllers with timing jitter. Last year we developed a method for truncating the coefficients of a linear controller while guaranteeing that a given set of relaxed performance constraints is met. The method sequentially and greedily truncates individual coefficients, using a Lyapunov certificate, typically in linear matrix inequality (LMI) form, to guarantee performance. This year we considered a continuous-time linear system with sampled constant linear state-feedback control and a convex quadratic performance measure. The sample times, however, were subject to variation within some known interval. Built on the previous results, we used the LMI method to derive a Lyapunov function to establish an upper bound on performance degradation due to the timing jitter. The same Lyapunov function was also used in a heuristic for finding a bad timing jitter sequence, which gave a lower bound on the possible performance degradation. Numerical experiments showed that these two bounds were often close, which meant that our bound is tight. We showed how LMI methods can be used to synthesize a constant state-feedback controller that minimizes the performance bound, for a given level of timing jitter.

Timing properties of embedded control systems. We have extended a convex optimization approach to bound the performance degradation of control systems introduced by Boyd to control systems with nondeterministic time-varying network loop delay. We consider a linear time-invariant continuous-time plant connected over a communication network to a remote controller. The network introduces bounded but time-varying delays between the controller and plant. We establish an upper bound on the worst-case performance degradation due to the network delays as a function of the delay bound, which can be used as a design parameter for the networked implementation. Numerical simulation results illustrate the degree of conservativeness of the bounds.

3.2 Model-Based Software Design and Verification

3.2.1 Model-Integrated Computing (Sztipanovits, Karsai, Volgyesi)

Decoupling abstraction layers. Model - based software design progresses along abstraction layers (design platforms) capturing essential design concerns. Effectiveness of the model-based design largely depends on how much the design concerns (captured in the abstraction layers) are orthogonal, i.e., how much the design decisions in the different layers are independent. Heterogeneity of embedded systems causes major difficulties in this regard. The controller dynamics is typically designed without considering implementation side effects (e.g. numeric accuracy of computational components, timing accuracy caused by shared resource and schedulers, time varying delays caused by network effects, etc.). Compositionality in one layer depends on a web of assumptions to be satisfied by other layers.

We have investigated theories and techniques for decoupling stability properties of networked controller dynamics from the effects of time-varying delays caused by networks and schedulers using two fundamentally different methods: passivity, and Timed Triggered Architecture (TTA). By imposing passivity constraints on the component dynamics, the design becomes insensitive to network effects, thus establishing orthogonality (with respect to network effects) across the controller design and implementation design layers. TTA uses architectural restrictions and static structure to achieve strict synchrony. We have experimented with both approaches (and with their combination) and evaluated their benefits.

3.2.2 Embedded Software Composition Platform (Lee)

We focused on exploring automatic code generation for parallel and distributed architectures. The two main types of parallel platforms are shared-memory and message passing. We set out to extend our ability to program these platforms, enabling users to explore the design space easily while preserving a high-level understandable programming model. We prototyped two new facilities in the current Ptolemy II code generator to generate Pthread, a user-level thread library, and MPI code. We can generate parallel code that targets different platforms from higher level specifications which allow for quick development and prototyping of parallel applications. This framework will allow users to parameterize several design choices such as the number of cores, targeting library, and partitioning of the application to quickly generate executable parallel code for comparison and tuning. We further extend this framework to insert profiling and feedback code into the generated program. This allows users to obtain execution traces and statistics, which can be fed back to the code generator to further tune and optimize the to produce better code. We implemented this code generation framework on top of the Ptolemy II framework, which is a heterogeneous modeling and simulation environment designed to allow users to explore high level models of computation. Currently, both the Pthread and MPI code generation engines have been implemented and are able to generate code from Process Network models.

Work on PTIDES (Programming Temporally Integrated Distributed Embedded Systems), which leverages time synchronization over distributed platforms, continues. Researchers from the University of Salzburg have implemented PTIDES the Timing Definition Language in Ptolemy. We are also collaborating with IBM on work using Exotasks to implement PTIDES.

3.2.3 Automated Source Code Verification and Testing (Clarke, Krogh)

We continue to develop new abstraction and iterative/adaptive refinement techniques to verify correctness of control software and hybrid dynamic systems. We are currently developing tools to demonstrate and evaluate the effectiveness of these methods for design-time verification. We have also developed a quantitative certificate approach to design of control systems with network delays based on convex optimization and initiated work on verification of translators for auto-code generation from design models. Finally, we have begun investigating statistical methods to perform probabilistic verification of discrete and hybrid systems.

Ongoing work on verifying Simulink models of nonlinear systems using Sensitivity Analysis. We initiated an adaptation of the reachability analysis technique using sensitivity analysis developed in the thesis of A. Donzé to the case of continuous-time nonlinear Simulink models. We use the Real Time Workshop toolbox of The Mathworks to generate code from the Simulink model and reuse this code in conjunction with a specific numerical solver to perform the sensitivity analysis. The method computes simulation traces and their sensitivity to parameters to estimate reachable tubes around trajectories. An automatic refinement of the set of parameters guarantees the coverage of the reachable set. It can efficiently find counterexamples and identify safe ranges of parameters for arbitrary nonlinear continuous systems. We are currently extending our implementation to handle models with discrete states.

Computing differential invariants of hybrid systems as fixedpoints. We introduced a fixedpoint algorithm for verifying safety properties of hybrid systems with differential equations that have right-hand sides that are polynomials in the state variables. In order to verify nontrivial systems without solving their dfferential equations and without numerical errors, we use a continuous generalization of induction, for which our algorithm computes the required differential invariants. As a means for combining local differential invariants into global system invariants in a sound way, our fixedpoint algorithm works with a compositional verification logic for hybrid systems. To improve the verification power, we further introduced a saturation procedure that refines the system dynamics successively with differential invariants until safety becomes provable. By complementing our symbolic verification algorithm with a robust version of numerical falsification, we obtain a fast and sound verification procedure.

Statistical model checking of mixed-analog circuits. Model checking properties for systems involving continuous state variables is known to be a difficult problem. This holds, in particular, for mixed-signal circuits, i.e., circuits for which there is an interaction between analog (continuous) and digital (discrete) quantities. We investigated the use of statistical model checking techniques for the analysis of mixed-signal circuits. Instead of verifying a property exhaustively with respect to the behaviors of the model, we evaluate it on a representative subset of behaviors, generated by simulation, and answer the question of whether the circuit satisfies the property with a probability greater than or equal to some value. The answer is correct up to a certain probability of error, which can be pre-specified. The method automatically determines the maximal number of simulations needed to achieve the desired accuracy, thus providing a convenient way to control the trade-off between precision and computational cost, even for complex systems. We provided a logic adapted to the specification of properties for mixed-signal circuits, in the temporal domain as well as in the frequency domain, which is highly relevant in this specific context. The applicability of the method was demonstrated on a model of a Delta-Sigma modulator for which previous formal verification attempts were too conservative and required excessive computation time.

3.3 Composable Tool Architectures

3.3.1 Advanced Open Tool Integration Framework (Karsai, Sztipanovits)

Formal specification of behavioral semantics. We have continued our efforts on the formal specification of behavioral semantics for domain specific modeling languages. In the last year we have experimented with using the DeVS formalism for behavioral specification and developed a time triggered scheduler as a refinement of the formal specification. We have made tradeoff studies between using the DeVS and the Abstract State Machine (ASM) formalisms.

3.3.2 Prototype Tool Chain (Volgyesi, Karsai, Sztipanovits)

Prototype toolchain. We continued our work on the prototype tool chain, FRED. The current architecture of the tool chain is show in Figure 1. The tool chain is capable to work with high level (controller) models from the MATLAB/Simulink environment (**MDL2MGA** tool), partition and assign components to nodes and tasks (ECSL-DP domain specific modeling language and the GME modeling environment) and generate code and runtime configuration for different distributed platforms (TTP/C, Linux, FreeRTOS). The code is generated in two steps; first the abstract syntax tree of the code is built (**SL/SF CodeGen** tools, SFC domain specific modeling language), then the actual C/C++/Java code is printed from the abstract model (**SFCPrint** tool). The most important benefits are the relatively low cost of adding support for additional programming languages and high level access to the executable code for external tools (eg. source code verification)

In the past year we added code generation support for the TTP/C platform and integrated the tool chain with the TTP Tools design environment. The **TTP CodeGen** tool generates glue code for encapsulating the platform independent source code in tasks on the TTP/C platform. Also, it translates the component and platform models into a proprietary database used by the TTP Tools to generate proper task and message schedule along with driver code on this platform.

We extended our set of experimental target platforms by building a time-triggered task scheduler and communication framework on top of the standard Linux POSIX threads and IP socket libraries (**FRODO**). This relatively lightweight kernel provides an open and low cost experimental time triggered environment. Code generation support for this platform is under development, but an offline (static) message and task scheduler tool is already available and is used for configuring executables on this platform. The scheduler tool is able to generate periodic task and message schedules based on rate, latency and data dependency requirements.

Recently, we have created a time triggered execution framework on Atmel AVR embedded controllers building on the FreeRTOS environment. This is a truly embedded (and resource constrained) platform, which will be used to run the control software for the Stanford STARMAC aircraft in our real-time simulation environment (see Section 3.4). Currently, this embedded kernel is capable to execute periodic tasks according to a static schedule (the same offline scheduler tool is used for computing the schedule), but does not support communication primitives.

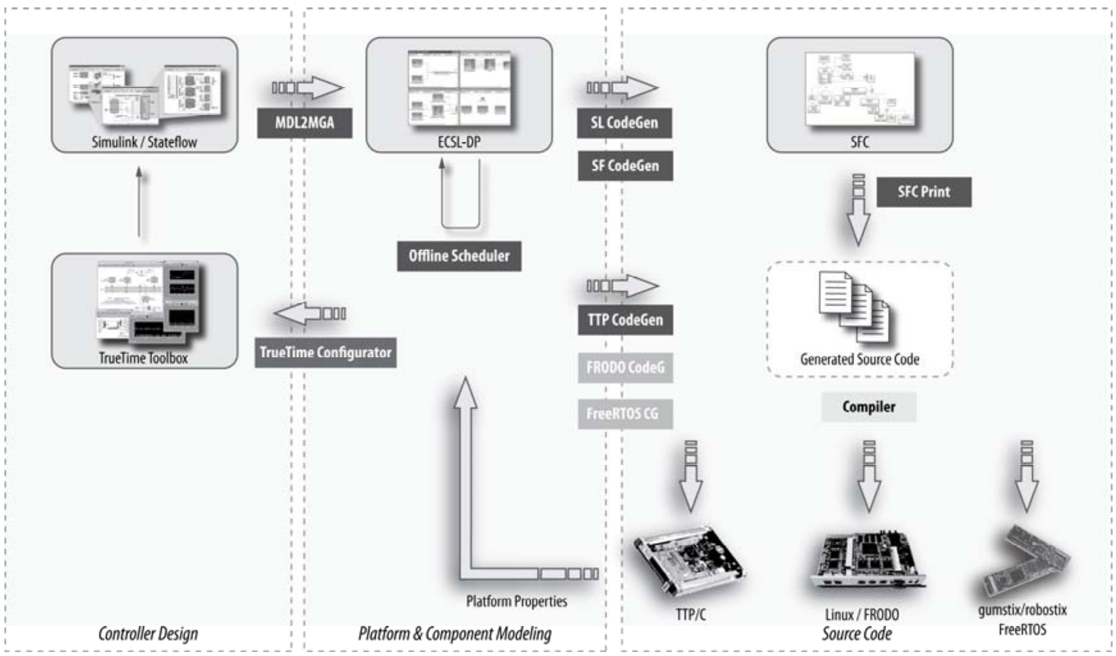

Figure 1: Current architecture of the FRED tool chain

We integrated the TrueTime simulation toolbox with the tool chain. This feature enables us to simulate the componentized, distributed and statically scheduled controller logic and evaluate various platform specific effects (jitter, latency, resource utilization, etc.). The TrueTime Configurator tool takes the component and platform models and builds up a similar architecture with TrueTime blocks in the Simulink environment. The generated blocks (computing nodes, communication channels) reuse and execute parts of the high level Simulink controller model according to the rules and characteristics of the simulated platform, thus provide excellent feedback in the controller design environment.

The utilization of TrueTime for simulating platform specific effects proved to be a very useful approach. For this reason we started to work on a lightweight platform assignment tool which can be used directly in the MATLAB/Simulink environment. Here, the platform assignment is a simple map (using Excel tables) from dataflow nodes to processors and from port nodes to networks. Assignments may be made with minimal consideration of the interconnections in the dataflow graph, as consistency can be checked prior to model generation.

3.4 Testing and Experimental Validation (Tomlin, Sastry, Lee, Karsai)

We continued testing the baseline controller design of the UAV platforms on the emerging model-based design tool suite.

Ptolemy tool suite. The Ptolemy II code generation tool was successfully used to implement a hill climbing algorithm using the same processor as is used on the STARMAC. Ptolemy II has also been extended to generate NuSMV Verification code. NuSMV is based on SMV, a symolic model checker created by CMU Model Checking team working on this MURI. We are addressing the difficulty that designers of embedded software systems face when doing formal verification. Existing theories and practices in verification are powerful, but when applying formal techniques, the use of detailed mathematical model descriptions in verification greatly increase the burden on system designers; construction of such models may be time consuming and error prone. We lay the groundwork for solving this problem by providing a mapping from actor models to mathematical models suitable for verification; the conversion is automatic with minimal human intervention. Meanwhile, the interactions between the verification model and its environment can guide us in designing how the implementation model interprets the raw data from sensors and to actuators, allowing us to reuse the verification model as the basis of its implementation model. Following these strategies, the productivity of designers and the correctness of designs can be maintained simultaneously.

We have started to construct a real-time simulation environment for the Stanford STARMAC quadrotor aircraft control software. Currently, the physical environment (flight dynamics, sensors, actuators) are simulated on a dedicated xPC computer. The most important state variables are displayed on a network connected monitoring workstation. We begun to integrate the generated controller software (running on the AVR platform) with the environment simulator. The control software is generated and configured with the FRED tool chain.

4. Personnel Supported

Vanderbilt:

1. Professor Janos Sztipanovits (PI)
2. Professor Gabor Karsai
3. Peter Volgyesi (research scientist)
4. Joe Porter (Graduate Student, funded by this contract)
5. Ryan Thibodeaux (Graduate Student, funded elsewhere)

Associated but not supported:

1. Himansu Neema (Senior Engineer)
2. Sandeep Neema (Senior Research Scientist)
3. Harmon Nine (Senior Engineer)
4. Graham Hemingway (Graduate Student)
5. Peter Humke (Graduate Student)

Berkeley:

1. Professor Claire Tomlin
2. Professor Edward A. Lee (Faculty, funded elsewhere)
3. Professor Shankar Sastry
4. Humberto Gonzales (Graduate Student, funded by this contract)
5. Gabe Hoffmann (Graduate Student at Stanford, funded elsewhere)
6. Gang Zhou (Graduate Student, funded by this contract)
7. Man-kit (Jackie) Leung (Graduate Student, funded by this MURI)
8. Issac Liu (Graduate Stuent, funded elsewhere)
9. Jia Zhou (Graduate Student, funded elsewhere)
10. Christopher Brooks (Software Engineer, funded 25%)
11. Jonathan Sprinkle (Research Associate, funded elsewhere)

CMU

1. Professor Bruce Krogh
2. Professor Edmund Clarke
3. Himanshu Jain, PhD candidate, Computer Science Dept., CMU
4. Flavio Lerda, PhD candidate, Computer Science Dept., CMU
5. Ajinkya Y. Bhave, PhD candidate, Dept. of ECE, CMU
6. Hitashyam Maka, MS candidate, Dept. of ECE, CMU

Associated but not supported:

5. Sumit Jha, PhD candidate, Computer Science Dept., CMU
6. Stephen Magill, PhD candidate, Computer Science Dept., CMU
7. Bryant Lee, PhD candidate, Computer Science Dept., CMU
8. Nishant Sinha, PhD candidate, Computer Science Dept., CMU
9. Constantinos Bartzis, Post Doc, Computer Science Dept., CMU
10. Tamir Heyman, Post Doc, Computer Science Dept., CMU
11. Azideh Farzan, Post Doc, Computer Science Dept., CMU
12. Silke Wagner, Post Doc, Computer Science Dept., CMU
13. Alexandre Donze, Post Doc, Computer Science Dept., CMU
14. James Kapinski, Post Doc, Dept. of ECE, CMU
15. Ingo Feinerer, Visiting Researcher, Computer Science Dept., CMU
16. Stacey Ivol, MS candidate, Dept. of ECE, CMU
17. Goran Freshe, Visiting Researcher, Dept of ECE, CMU

Stanford

1. Professor Stephen P. Boyd,
2. Jo¨elle Skaf, Ph.D. Candidate
3. Siddharth Joshi, Ph.D. Candidate
4. Almir Mutapcic, Ph.D. Candidate
5. Seung Jean Kim, Consulting Professor

5. Publications

1. Man-Kit Leung and Edward A. Lee, "An Extensible Software Synthesis Framework for Heterogeneous Actor Models," in Proceedings of the 7th Model-driven High-level Programming of Embedded Systems Workshop (SLA++P 08), Work-in-Progress Session, Budapest, Hungary, March 2008. http://chess.eecs.berkeley.edu/pubs/401.html

2. Patricia Derler, Thomas Huining Feng, Edward A. Lee, Slobodan Matic, Hiren Patel, Yang Zhao, Jia Zou. "PTIDES: A Programming Model for Distributed Real-Time Embedded Systems". RTSS'08, submitted, May, 2008.

3. Edward A. Lee. "Time is a Resource, and Other Stories". International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC), May, 2008; Position statement for panel: "Wrong Assumptions and Neglected Areas in Embedded Research".

4. Toward an Effective Execution Policy for Distributed Real-Time Embedded Systems, Thomas Huining Feng, Edward A. Lee, Hiren Patel, Jia Zou, Proceedings of the 14th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 08), Work-in-Progress Session, April, 2008

5. Christopher Brooks, Thomas Huining Feng, Edward A. Lee, Reinhard von Hanxleden. "Multimodeling: A Preliminary Case Study". Technical report, EECS Department, University of California, Berkeley, UCB/EECS-2008-7, January, 2008.

6. J. Ding, C. Fischione, A. Sangiovanni-Vincentelli, C. J. Tomlin: "Design of Wireless Communication Parameters for Collision Avoidance on a UAV Platform," Submitted to the IEEE Conference on Decision and Control, February 2008.

7. J. Ding, J. Sprinkle, S. S. Sastry and C. J. Tomlin: "Reachability analysis for an Automatic Refueling Protocol," Submitted to the IEEE Conference on Decision and Control, February 2008.

8. G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin: "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment," Proceedings of the AIAA Guidance, Navigation, and Control Conference, Hilton Head, AIAA Paper Number 2007-6461, August 2007.

9. G. M. Hoffmann, D. Gorinevsky, R. W. Mah, C. J. Tomlin, and J. D. Mitchell: "Fault Tolerant Relative Navigation using Inertial and Relative Sensors," Proceedings of the AIAA Guidance, Navigation, and Control Conference, Hilton Head, AIAA Paper Number 2007-6789, August 2007.

10. S. L. Waslander and C. J. Tomlin: “Convergence of lump-sum markets with priceanticipating agents,” Proceedings of the AACC American Control Conference, New York, July 2007.

11. Holger Giese, Gabor Karsai, Edward Lee, Bernhard Rumpe, and Bernhard Schatz, Model-based Engineering of Embedded Real-time Systems, Dagstuhl Seminar Proceedings 07451, Internationales Begegnungs- und Forschungszentrum fur Informatik (IBFI), Schloss Dagstuhl, Germany, Nov. 5-9, 2007

12. Joelle Skaf and Stephen Boyd: “Controller Coefficient Truncation Using Lyapunov Performance Certificate,” Proceedings of the European Control Conference, July 2007

13. Stephen Boyd: “Hyperspectral Image Unmixing via Alternating Projected Subgradients,” Proceedings Asilomar Conference, November 2007

14. Stephen Boyd: “A Tractable Method for Robust Downlink Beamforming in Wireless Communications,” Proceedings Asilomar Conference, November 2007

15. Stephen Boyd: “Dynamic Network Utility Maximization with Delivery Contracts,” To appear in IFAC World Congress, July 2008.

16. Mutapcic, S.-J. Kim, and S. Boyd: “Robust Chebyshev FIR Equalization,” Proc. 50th IEEE Global Communications Conference (GLOBECOM ’07), pages 3074-3079, Washington DC, Nov. 2007

17. S. Samar, S. Boyd, and D. Gorinevsky, “Distributed estimation via dual decomposition”, Proceedings of European Control Conference, Kos, Greece, July 2007.

18. Murali, A. Mutapcic, D. Atienza, R. Gupta, S. Boyd, and G. De Micheli: “Temperature Aware Processor Frequency Assignment for MPSoCs Using Convex Optimization,” Proc. 5th Int. Conf. on Hardware/Software Codesign and System Synthesis (CODES+ISSS), pages 111-116, Salzburg, Austria, Oct. 2007

19. Azadeh Farzan, Y. Chen, E. M. Clarke, Y. Tsan, B. Wang: “Extending Automated Compositional Verification to the Full Class of Omega-Regular Languages,” To appear in Proceedings of Tools and Algorithms for the Construction and Analysis of Systems (TACAS) 2008.

20. Flavio Lerda, James Kapinski, Hitashyam Maka, Edmund M. Clarke, Bruce H. Krogh: “Model Checking In-The-Loop,” To appear in Proceedings of the 2008 American Control Conference.

21. Flavio Lerda, James Kapinski, Edmund M. Clarke, and Bruce H. Krogh: “Verification of Supervisory Control Software Using State Proximity and Merging,” To appear in Pro-

ceedings of the 11th International Conference on Hybrid Systems: Computation and Control.

22. Kai Chen, Janos Sztipanovits, Sandeep Neema: “A Case Study on Semantic Unit Composition,” *Proceedings of International Workshop on Modeling in Software Engineering*, ICSE 2007, Minneapolis MN, May 20-26, 2007

23. Ethan Jackson and Janos Sztipanovits, “Constructive Techniques for Meta- and Model level Reasoning,” *Proceedings of MODELS07*, Nashville, TN, October 2-5, 2007

24. Ryan Thibodeaux and Gabor Karsai. “Model-Based Specification and Implementation of a Model of Computation,” Fourth European Conference on Model-Driven Architecture – Foundations and Applications, Berlin, Germany, June 2008.

25. Gabor Karsai and Ryan Thibodeaux. “Modeling Models of Computations,” Workshop on Modeling, Validation and Heterogeneity at the IEEE International Conference on Software Testing, Verification, and Validation, Lillehammer, Norway, April 2008.

26. Himanshu Jain, Daniel Kroening, Natasha Sharygina, Edmund Clarke: “Word Level Predicate Abstraction and Refinement for Verifying RTL Verilog,” To appear in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2007

27. G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, "STARMAC: The Stanford Testbed of Autonomous Rotorcraft for Multiple Agent Control", in preparation for submission to the AIAA Journal of Guidance, Control, and Dynamics. February, 2008

28. M. K. Oishi, I. M. Mitchell, A. M. Bayen and C. J. Tomlin: “Hybrid system verification: application to user-interface design,” Accepted to appear in the IEEE Transactions on Control Systems Technology, 2007.

29. J. Hu, M. Prandini, and C. Tomlin: “Conjugate points in formation constrained optimal multi-agent coordination: a case study, “ SIAM J. Control and Optimization, vol. 45, no. 6, pp. 2119-2137, 2007.

30. K.-L. Hsiung, S.-J. Kim, and S. Boyd Tractable Approximate Robust Geometric Programming Optimization and Engineering, published online October 2007.

31. J. Skaf and S. Boyd: “Filter Design with Low Complexity Coefficients,” To appear in IEEE Transactions on Signal Processing.

32. M. Lobo, M. Fazel, and S. Boyd: “Portfolio Optimization with Linear and Fixed Transaction Costs,” Annals of Operations Research, 152(1):376-394, July 2007.

33. S.-J. Kim, S. Boyd, S. Yun, D. Patil, and M. Horowitz: “A Heuristic for Optimizing Stochastic Activity Networks with Applications to Statistical Digital Circuit Sizing,” Opti-

mization and Engineering, 8(4):397-430, December 2007

34. L. Xiao, S. Boyd, and S.-J. Kim: “Distributed Average Consensus with Least-Mean Square Deviation,” Journal of Parallel and Distributed Computing, 67(1):33-46, 2007.

35. S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky: “A Method for Large-Scale $\ell 1$-Regularized Least Squares,” IEEE Journal on Selected Topics in Signal Processing, 4(1):606-617, December 2007.

36. Ghosh, S. Boyd and A. Saberi: “Minimizing Effective Resistance of a Graph,” SIAM Review, problems and techniques section, 50(1):37-66, February 2008.

37. M. Grant and S. Boyd: “Graph Implementations for Nonsmooth Convex Programs,” Recent Advances in Learning and Control (tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors, pages 95-110, Lecture Notes in Control and Information Sciences, Springer, 2008.

38. Yi Ma, Rene Vidal and Shankar Sastry, "Generalized Principal Component Analysis”, Springer Verlag to appear 2009.

39. Ryan Thibodeaux. “The Specification and Implementation of a Model of Computation.” Master’s Thesis, Vanderbilt University, EECS Dept, Nashville, TN, May 2008

40. A. Donzé, T. Dang, O. Maler, N. Shalev, “Sensitive State Space Exploration,”, submitted to the *2008 Conference on Decision and Control*.

41. E. M. Clarke, A. Donzé and A. Legay “Model Checking of Mixed-Analog Circuits,” Presented at *FAC'08 (Formal Verification of Analog Circuits), a satellite workshop at CAV 2008.*

42. Himanshu Jain, Edmund M. Clarke, Orna Grumberg, “Efficient Craig Interpolation for Linear Diophantine (Dis) Equations and Linear Modular Equations,”. CAV 2008, To appear in *20th International Conference on Computer Aided Verification.*

43. J. Kapinkski, A. Donzé, F. Lerda, H. Maka, S. Wagner, E. M. Clarke, and B. H. Krogh, "Control Software Model Checking Using Bisimulation Functions for Nonlinear Systems,". Submitted to the *2008 Conference on Decision and Control*.

44. A. Bhave and B. H. Krogh, “Performance Bounds on State-Feedback Controllers with Network Delay,”. Submitted to the *2008 Conference on Decision and Control*.|

45. A. Donzé, F. Lerda, H. Maka, E. M. Clarke, and B. H. Krogh, "Model Checking Control Software for Nonlinear Dynamic Systems," Extended abstract submitted to invited sessions of the *2008 Guidance, Navigation, and Control Conference*.

46. Flavio Lerda, James Kapinski, Hitashyam Maka, Edmund M. Clarke, Bruce H. Krogh, “Model Checking In-The-Loop.” In *Proceedings of the 2008 American Control Conference.*

47. Flavio Lerda, James Kapinski, Edmund M. Clarke, and Bruce H. Krogh, “Verification of Supervisory Control Software Using State Proximity and Merging.” In *Proceedings of the 11th International Conference on Hybrid Systems: Computation and Control, 2008.*

48. Frehse, S. K. Jha, B. H. Krogh, “A Counterexample-Guided Approach to Parameter Synthesis for Linear Hybrid Automata,” G, in *Proceedings of the 11th International Conference on Hybrid Systems: Computation and Control, 2008.*

49. Azadeh Farzan, Y. Chen, E. M. Clarke, Y. Tsan, B. Wang, “Extending Automated Compositional Verification to the Full Class of Omega-Regular Languages.”. To appear in *Proceedings of Tools and Algorithms for the Construction and Analysis of Systems (TACAS) 2008*.

50. Himanshu Jain, Daniel Kroening, Natasha Sharygina, Edmund Clarke , “VCEGAR: Verilog CounterExample Guided Abstraction Refinement.” . *TACAS 2007, 13th International Conference on Tools and Algorithms for the Construction and Analysis of Systems.*

51. Himanshu Jain, Daniel Kroening, Natasha Sharygina, Edmund Clarke, “Word Level Predicate Abstraction and Refinement for Verifying RTL Verilog.” (IEEE TCAD 2007), To appear in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.*

52. Xenofon Koutsoukos, Nicholas Kottenstette, Joe Hall, Panos Antsaklis, Janos Sztipanovits , Passivity-Based Control Design of Cyber-Physical Systems", *Proceedings of the Workshop on Cyber-Physical Systems - Challenges and Applications (CPS-CA'08)* held in conjunction with In conjunction with the 4th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS'08)

53. Nicholas Kottenstette, Xenofon Koutsoukos, Joe Hall, Panos Antsaklis, Janos Sztipanovits, “ Passivity-Based Design of Wireless Networked Control Systems for Robustness to Time-varying Delays", *Submitted paper*

54. Karsai G., Narayanan A., "On the Correctness of Model Transformations in the Development of Embedded Systems", Lecture Notes in Computer Science, vol. 4888: Springer, 2007.

55. Balasubramanian D., Narayanan A., Neema S., Shi F., Thibodeaux R., Karsai G., "A Subgraph Operator for Graph Transformation Languages", Electronic Communications of the EASST, vol. 6, 2007.

56. Balasubramanian D., Narayanan A., Neema S., Ness B., Shi F., Thibodeaux R., Karsai G., "Applying a Grouping Operator in Model Transformations", The Third International

Workshop and Symposium on Applications of Graph Transformation with Industrial Relevance (AGTIVE), Kassel, Germany, 10/2007.

57. Narayanan A., Karsai G., "Specifying the Correctness Properties of Model Transformations", 3rd International Workshop on Graph and Model Transformation (GraMoT), 05/2008.

58. Narayanan A., Karsai G., "Towards Verifying Model Transformations", Electronic Notes in Theoretical Computer Science, vol. 211, pp. p.191-200, 2008.

59. Narayanan A., Karsai G., "Verifying Model Transformations by Structural Correspondence", Electronic Communications of the EASST, vol. 10, 2008.

60. Jackson, E., Sztipanovits, J.: 'Formalizing the Structural Semantics of Domain-Specific Modeling Languages," *Journal of Software and Systems Modeling* (to be published in 2008)

6. Interactions/Transitions

6.1 Participation/presentations at meetings, conferences, seminars

1. MURI team attended the bi-weekly MURI telcons.

2. HCDDES Review Meeting, September 6, 2007, Berkeley.
   Edward Lee presented "Principled Design of Embedded Software"
   Claire Tomlin and Shanka Sastry presented "Robust Hybrid and Embedded Systems Design for Quadrotor Platform"
   Gabor Karsai presented "Toward a Model-Based Tool Chain for High Confidence Design"
   Janos Sztipanovits presented "Project Overview"
   Stephen Boyd presented "Controller Coefficient Truncation Using Lyapunov performance Certificate"
   Bruce Krogh presented "Model-based Testing and Verification of Embedded System Implementations"
   Edmund Clarke presented "Automated Source Code Verification and Testing"

3. Technical Interchange meeting, October 22-26, 2007, Vanderbilt
   Day 1: The FRED Tool chain (Vanderbilt)
   Day 2: Verification (CMU)
   Day 3: Execution Platforms (Vanderbilt, Berkley)
   Day 4: Tool Integration (Vanderbilt, Berkeley)

4. Workshop: From Embedded Systems to Cyber-Physical Systems: a Review of the State-of-the-Art and Research Needs
   Sanjit A. Seshia, Edward A. Lee, Claire Tomlin presented "Teaching Embedded Systems

to Berkeley Undergraduates: EECS124 at Berkeley"
Edward A. Lee presented "Making Time Essential in Computation"
Janos Sztipanovits presented "Cyber Physical Systems: New Challenges for Model-based Design”

5. International Conference on Hybrid Systems Computation and Control 2008, April 22-24, 2008, St. Louis. Edmund C. Clarke, James Kapinski, and Bruce H. Krogh: Verification of Supervisory Control Software Using State Proximity and Merging.

6. International Conference on Hybrid Systems Computation and Control 2008, April 22-24, 2008, St. Louis. Goran Freshe, Sumit Jha, Bruce H. Krogh: A Counterexample-Guided Approach to Parameter Synthesis for Linear Hybrid Automata.

7. Workshop on Real Time Control of Hybrid Systems, Budapest, Hungary, October 29, 2007. Bruce H. Krogh and James Kapinski: Model Checking Embedded Control System Designs.

8. 2008 American Control Conference, June 11 - 13, 2008, Seattle Washington. Bruce H. Krogh: Model Checking In-The-Loop.

9. Systems Research Center Seminar Series, University of Maryland, May 2, 2008. Bruce H. Krogh: Applications of Formal Methods in Model-Based Development of Embedded Control Systems.

10. The 6th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS08), September 15—17, 2008, Salzburg, Austria. Bruce H. Krogh: From Analysis to Design

11. International Workshop on Hybrid Systems: Modeling, Simulation And Optimization on May 14-16, 2008, Istanbul, Turkey. Bruce H. Krogh: Iterative Relaxation Abstraction for Verification and Design of Hybrid Systems.

12. Man-Kit Leung presented his paper at SLA++P, Budapest, Hungary

13. Joseph Porter, Graduate student at Vanderbilt visited Prof. Stephen Boyd’s Lab for three weeks in May, 2008.

6.2 Consultative and advisory functions to other laboratories and agencies, especially Air Force and other DoD laboratories. Provide factual information about the subject matter, institutions, locations, dates, and names(s) of principal individuals involved

1. Janos Sztipanovits:
    a. Vice-chair of the S&T Review panel of AFRL/RI in October 2007.
    b. Executive Review of the AFRL Flight Critical Systems Software Initiative (FCSSI); Air Force Research Laboratory, Air Vehicles Directorate, 30 April 2008 David Homan

AFRL, Control Systems Development and Applications
david.homan@wpafb.af.mil

c. Study Chair of the AF SAB FY08 Study on "Defending and Operating in a Cyber Contested Environment"

2. Edward A. Lee:
   a. Air Force Research Laboratory, AFRL/RIEA, Rome, NY
      Michael Manno
      michael.manno@rl.af.mil
      (315) 330-7517=20

      The objective of the Extensible Modeling and Analysis Framework (EMAF) effort is tobuild on top of Ptolemy II and adapt Ptolemy II for the rapid construction and configuration of modeling and analysis systems that incorporate disparate technologies. The purpose of this gap-filling project is to develop technologies for future incorporation into large-scale modeling and analysis systems, with specific focuses on scalable algorithm description, composition of heterogeneous components, and synthesis of efficient deployable decision-support systems that exploit multicore and distributed computing platforms.
      In particular, we have applied the code generation infrastructure developed under this MURI to a very large problem consisting of roughly 13000 actors. We were able to reduce the run time from roughly 10 minutes to 3 seconds.

   b. Lockheed Martin Advanced Technology Laboratory
      Trip Denton
      ldenton@atl.lmco.com
      3 Executive Campus, 6th Floor;Cherry Hill, NJ, 08002, USA
      Work: 856 792-9071 fax:856 792-9925

      NAOMI Project (http://chess.eecs.berkeley.edu/naomi)
      (Also participating are Vanderbilt and UIUC)
      The purpose of the NAOMI project is to allow disparate modeling tools to be used to ether by tracking model changes within each system where a particular tool owns attributes of the overall design and provides attribute changes to other tools. The NAOMI project may result in useful technology that will allow easier collaboration on this MURI project. This project is using pedestrian/automobile traffic lights as a design driver. We have integrated Ptolemy II to the Naomi framework, which allows different tools to own attributes and update other tools when changes occur to those attributes.
      We have transferred models that use graph transformation and event relationship graphs.

   c. The US Army Research Laboratory
      Jeff DeHart, jdehart@arl.army.mil
      Scalable Composition of Systems (SCOS)
      http://chess.eecs.berkeley.edu/scos

The objective of the SCOS research project is to provide scalable techniques for the composition of subsystems in a system-of-systems (SoS) framework for large, complex applications such as FCS.

SCOS has synergy with this MURI project in that it deals with large systems. In particular:
- we are using the EmbeddedCActor to wrap legacy C code
- we are collaborating on work on the Kepler Project
- we are using Graph Transformations on models

6.3 Technology Assists, Transitions, and Transfers.

1. Ptolemy II 7.0.1 was released in April, 2008. Ptolemy II includes the code generation facility. The Ptolemy source tree is available via Subversion. The Ptolemy group has had the following significant, on-going interactions:

   BOSCH Research Center, Palo Alto
   The Ptolemy Hierarchical Orthogonal Multi-Attribute Solver (PtHOMAS) project which is using the Ptolemy Type system to analyze properties of a model such as const-ness of signals.

   HSBC Bank, London
   Ongoing work involving a Ptolemy GUI (Triquetrum.org) and performance analysis.

2. Vanderbilt's MIC tool suite (GME, GReAT, UDM, OTIF) had one major release during the last year. The released tools are available through the ESCHER and ISIS download sites
3. Vanderbilt continued working with GM, Raytheon and BAE Systems research groups on transitioning model-based design technologies into programs.
4. Vanderbilt continued working with Boeing's FCS program on applying the MIC tools for precise architecture modeling and systems integration

6.4 New discoveries, inventions, or patent disclosures.

None.

6.5 Honors and Awards

1. Edmund C. Clarke: A.M. Turing Award, 2008
2. Edmund M. Clarke named the 2008 Herbrand Award recipient
3. Edmund M. Clarke named University Professor at Carnegie Mellon University
4. Claire Tomlin:

   a. Chancellor's Professorship of EECS, UC Berkeley (2007-2010)
   b. Plenary Speaker, International Conference on Systems Biology (ICSB), San Diego, October 2007.
   c. Plenary Speaker, IFAC Symposium on Nonlinear Control Systems (NOLCOS), Pretoria, August 2007.
5. Shankar Sastry:
   a. Appointed Dean of Engineering, UC Berkeley, July 2007
   b. Technical Advisory Group, President's Council on Science and Technology, 2006-07
   c. Plenary Speaker, ICASSP, Honolulu, April 2007
   d. Plenary Speaker, IEEE CASE 2007, Phoenix, Arizona, October 2007
   e. Plenary Speaker, Hybrid Systems Computation and Control, Pisa, March 2007
6. Janos Sztipanovits:
   a. Kai Chen, Janos Sztipanovits and Sandeep Neema, "Compositional Specification of Behavioral Semantics," in Proceedings of DATE 2007, Nice, France.
      i. Best Paper Award, DATE 2007
      ii. Selected for publishing in the Springer volume of *The Most Influential Papers of 10 Years of DATE*
   b. Keynote Speaker, "Domain Specific Modeling Languages: Making Semantics Explicit" *OMG Technical Workshop,* Jacksonville, FL, September 25, 2007
   c. Keynote Speaker, "Towards Systematic Model-based Development of Embedded Systems," *Workshop on Towards a Systematic Approach to Embedded Systems Design - DATE 07*, Nice, France, April 20, 2007
7. B.A.E.F Grant (Belgian American Educational Fundation). Amount: 30, 000 USD. Awarded to Axel Legay