ANNUAL REPORT

FOUNDATIONS OF HYBRID AND EMBEDDED SYSTEMS AND SOFTWARE

NSF/ITR PROJECT – AWARD NUMBER: CCR-0225610

UNIVERSITY OF CALIFORNIA, BERKELEY
VANDERBILT UNIVERSITY
UNIVERSITY OF MEMPHIS

June 19, 2007

PERIOD OF PERFORMANCE COVERED: JUNE 1, 2006 – MAY 31, 2007

Contents

1. Participants

1.1. People

PRINCIPAL INVESTIGATORS:

THOMAS HENZINGER (UC BERKELEY, EECS)
EDWARD A. LEE (UC BERKELEY, EECS)
ALBERTO SANGIOVANNI-VINCENTELLI (UC BERKELEY, EECS)
SHANKAR SASTRY (UC BERKELEY, EECS)
JANOS SZTIPANOVITS (VANDERBILT, ECE)
CLAIRE TOMLIN (UC BERKELEY, EECS)

FACULTY INVESTIGATORS:

AHMAD BAHAI (UC BERKELEY, EECS)
RUZENA BAJCSY (UC BERKELEY, EECS)
GAUTAM BISWAS (VANDERBILT, CS)
RASTISLAV BODIK (UC BERKELEY, EECS)
BELLA BOLLOBAS (MEMPHIS, MATHEMATICS)
JEROME A. FELDMAN (UC BERKELEY)
KENNETH FRAMPTON (VANDERBILT, ME)
J. KARL HEDRICK (UC BERKELEY, ME)
GABOR KARSAI (VANDERBILT, ECE)
KURT KEUTZER (UC BERKELEY, EECS)
T. JOHN KOO (VANDERBILT)
WAGDY H. MAHMOUD (TENNESSEE TECH. UNIVERSITY)
GEORGE NECULA (UC BERKELEY, EECS)
SRINI RAMASWAMY (TENNESSEE TECH. UNIVERSITY)
PRAVIN VARAIYA (UC BERKELEY, EECS)
MASAYOSHI TOMIZUKA (UC BERKELEY, ME)

POST DOCTORAL RESEARCHERS:

AKOS LEDECZI (VANDERBILT)
JONATHAN SPRINKLE (UC BERKELEY)

GRADUATE STUDENTS:

ALESSANDRO ABATE (UC BERKELEY)
AARON AMES (UC BERKELEY)
SAURABH AMIN (UC BERKELEY)
DANIEL BALASUBRAMANIAN (VANDERBILT)
ARINDAM CHAKRABARTI (UC BERKELEY)

DENNIS CHANG (UC BERKELEY)
KRISHNENDU CHATTERJEE (UC BERKELEY)
ELAINE CHEONG (UC BERKELEY)
ABHIJIT DAVARE (UC BERKELEY)
DOUGLAS DENSMORE (UC BERKELEY)
MATTHEW EMERSON (VANDERBILT)
AARONG FLEETWOOD (VANDERBILT)
THOMAS HUINING FENG (UC BERKELEY)
SUMITRA GANESH (UC BERKELEY)
ARKEDEB GHOSAL (UC BERKELEY)
GRAHAM HEMMINGWAY (VANDERBILT)
ETHAN JACKSON (VANDERBILT)
FARINAZ KOUSHANFAR (UC BERKELEY)
ALEXANDER KURZHANSKIY (UC BERKELEY)
CHRISTINA LEE (VANDERBILT)
JONGHO LEE (UC BERKELEY)
DAVID P. MANDELIN (UC BERKELEY)
SLOBODAN MATIC (UC BERKELEY)
ELEFTHERIOS MATSIKOUDIS (UC BERKELEY)
TREVOR MEYEROWITZ (UC BERKELEY)
BHARATHWAJ MUTHUSWARMY (UC BERKELEY)
TAKASHI NAGATA (UC BERKELEY)
ALESSANDRO PINTO (UC BERKELEY)
WILLIAM PLISHKER (UC BERKELEY)
VINAYAK PRABHU (UC BERKELEY)
KAUSHIK RAVINDRAN (UC BERKELEY)
JANOS SALLAI (VANDERBILT)
PANNAG SANKETI (UC BERKELEY)
TRIPTI SAXENA (VANDERBILT)
RYAN T THIBODEAUX (VANDERBILT)
GUOGIANG WANG (UC BERKELEY)
YANG YANG (UC BERKELEY)
JOSE CARLOS ZAVALA (UC BERKELEY)
HAIBO ZENG (UC BERKELEY)
YANG ZHAO (UC BERKELEY)
HAIYANG ZHENG (UC BERKELEY)
GANG ZHOU (UC BERKELEY)
YE ZHOU (UC BERKELEY)
QI ZHU (UC BERKELEY)

UNDERGRADUATE STUDENTS:

ZULHIMI AHMAD (VANDERBILT)
ANGELINE BROWN (VANDERBILT)
LOUISE COLLINS (VANDERBILT)
DOMINIQUE DUNCAN (UC BERKELEY/UNIVERSITY OF CHICAGO)

TARRELL EZELL (VANDERBILT)
DANIELLE JONES (VANDERBILT)
CHARLES LEFONT (VANDERBILT)
KENNETH MCPHERSON (VANDERBILT)
NANDITA MITRA (UC BERKELEY/RUTGERS)
NASHLIE SEPHUS (UC BERKELEY/MISSISSIPPI STATE)
HEATHER TAYLOR (UC BERKELEY/UNIVERSITY OF VERMONT)
STEPHANIE WRIGHT (VANDERBILT)

TECHNICAL STAFF, PROGRAMMERS:

GYORGY BALOGH (VANDERBILT)
CHRISTOPHER BROOKS (UC BERKELEY)
PHILLIP LOARIE (UC BERKELEY)
DAN STEWARD (VANDERBILT)
MARY STEWART (UC BERKELEY)
BRIAN WILLIAMS (UC BERKELEY)

BUSINESS ADMINISTRATORS:

ROBERT BOXIE (VANDERBILT, SIPHER COORDINATOR)
MICHELE M. CODD (VANDERBILT)
TRACEY RICHARDS (UC BERKELEY)

1.2. Partner Organizations:

UNIVERSITY OF CALIFORNIA, BERKELEY
VANDERBILT
MEMPHIS

1.3. Collaborators:

NORIYASU ADACHI (TOYOTA TECHNICAL CENTER)
LUCA DE ALFARO (UC SANTA CRUZ)
HUGO A. ANDRADE (NATIONAL INSTRUMENTS)
J. D. AXELROD (STANFORD)
ILKAY ALTINTAS (SDSC)
AARON AMES (CALTECH)
SHAMIK BANDYOPADHYAY (UC BERKELEY)
MARIA DOMENICA DI BENEDETTO (UNIVERSITA' DELL'AQUILA)
CHAD BERKLEY (UC SANTA BARBARA)
ALBERT BENVENISTE (INRIA)
DIRK BEYER (SIMON FRASER UNIVERSITY)
MARK BIGGIN (LBNL)

THOMAS BRIHAYE (LSV-CNRS & ENS DE CACHAN, FRANCE)
JEFF BURCH (AGILENT)
BENOIT CAILLAUD (INRIA)
LUCA CARLONI (COLUMBIA UNIVERSITY)
MINGHUA CHEN (MICROSOFT RESEARCH)
ADAM DONLIN (XILINX RESEARCH LABS)
LAURENT DOYEN (EPFL)
STEPHEN EDWARDS (COLUMBIA UNIVERSITY)
JOHN EIDSEN (AGILENT)
MIKE EISEN (LBNL)
J. MIKAEL EKLUND (UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY)
MARCO FAELLA (UC SANTA CRUZ)
ANTOON GODERIS (UNIVERSITY OF MANCHESTER)
JEFF GRAY (UNIVERSITY OF ALABAMA (BIRMINGHAM)
ROBERT D. GREGG (UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN)
ESTEN GROTLI (UNIVERSITY OF TRONDHEIM)
PAOLO GIUSTO (GM RESEARCH)
BRUCE HAMILTON (AGILENT)
DAN HIGGINS (UC SANTA BARBARA)
DANIEL IERCAN (POLITECHNICA U. OF TIMISOARA)
ALESSANDRO D'INNOCENZO (UNIVERSITA' DELL'AQUILA)
STAN JEFFERSON (AGILENT)
EFRET JAEGER (SDSC)
MATTHEW JONES (UC SANTA BARBARA)
WOOYOUNG JUNG (DGIST)
TOMOYUKI KAGA (TOYOTA TECHNICAL CENTER)
SRI KANAJAN (GM RESEARCH)
EUNJOO KIM (DGIST)
CHRISTOPH KIRSCH (UNIVERSITY OF SALZBURG)
DOMINIK LANGEN (INFINEON TECHNOLOGIES)
ELIZABETH LATRONICO (BOSCH)
MAN-KIT LEUNG (UC BERKELEY)
JIE LIU (MICROSOFT RESEARCH)
XIAOJUN LIU (SUN MICROSYSTEMS)
JOHN LYGEROS (ETH ZURICH)
BERTRAM LUDASCHER (UC DAVIS)
YI MA (UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN)
THOMAS MANDL (BOSCH)
RUPAK MAJUMDAR (UC LOS ANGELES)
JOSEPH MAKIN (UC BERKELEY)
RADU MARCULESCU (CMU)
RICK MCGEER (HP LABS)
ANDREW MIHAL (UC BERKELEY)
MARK MILLER (HP LABS)
MARCO DI NATALE (GENERAL MOTORS RESEARCH)

STEPHEN NEUENDORFFER (XILINX)
MARIA PARANDINI (POLITECNICO DI MILANO)
ROBERT PASSERONE (UNIVERSITY OF TRENTO ITALY)
CLAUDIO PINELLO (CADENCE BERKELEY LABS)
NIR PITERMAN (EPFL)
GIORDANO POLA (UNIVERSITA' DELL'AQUILA)
MARIA PRANDINI (POLITECNICO DI MILANO)
JEAN-FRANCOIS RASKIN (UNIVERSITE LIBRE DE BRUXELLES)
SANJAY REKHI (CYPRESS SEMICONDUCTOR)
MATTI ROSSI (UNIVERSITY OF JYVASKALA)
MIRKO SAUERMANN (INFINEON TECHNOLOGIES)
NADATHUR SATISH (UC BERKELEY)
JOSEPH SIFAKIS (VERIMAG GRENOBLE)
MYOUNGKYU SOHN (DGIST)
MARIELLE STOELINGA (UNIVERSITY OF TWENTE)
EELCO SCHOLTE (UNITED TECHNOLOGIES RESEARCH CENTER)
CAROLYN TALCOTT (SRI INTERNATIONAL)
ASHISH TIWARI (SRI INTERNATIONAL)
J.-P. TOLVANEN (META CASE)
STAVROS TRIPAKIS (CADENCE BERKELEY LABS)
BEN UPCROFT (UNIVERSITY OF SYDNEY)
RANDALL URBANCE (GM RESEARCH)
GUANQIANG WANG (EECS (UC BERKELEY)
LYNN WANG (UC BERKELEY)
ERIC D.B. WENDEL (SENSIS CORPORATION)
JOHN WRIGHT (UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN)
JOSEPH WYSOCKI (INDEPENDENT CONSULTANT (MALIBU (CALIFORNIA)
YEON-MO YANG (DGIST)
GUANG YANG (UC BERKELEY)
HAKAN YAZAREL (TOYOTA TECHNICAL CENTER)
AVIDEH ZAKOR (UC BERKELEY)
QI ZHU (UC BERKELEY)

2. Activities and Findings

2.1. Project Activities

This is the fifth Annual Report for the NSF Large ITR on "Foundations of Hybrid and Embedded Systems and Software." This year generally saw a great deal of synergy among various researchers. This research activity is primarily organized through CHESS at the University of California, Berkeley (Center for Hybrid and Embedded Systems and Software, http://chess.eecs.berkeley.edu), ISIS at Vanderbilt University (Institute for Software Integrated Systems, http://www.isis.vanderbilt.edu), and the Department of Mathematical Sciences, (http://msci.memphis.edu) at the University of Memphis.

The web address for the overall ITR project is:

http://chess.eecs.berkeley.edu/projects/ITR/main.htm

This web site has links to the proposal and statement of work for the project.

Main events for the ITR project in its fifth year were:

- CHESS NSF ITR Fourth Year Site Visit, October 4, 2006, Alexandria, VA. The program and presentations are available at http://chess.eecs.berkeley.edu/conferences/06/FallReview/index.htm
- CHESS Winter Meeting, February 14, 2007, UC Berkeley. The program and the presentations are available at http://chess.eecs.berkeley.edu/conferences/07/WinterReview/program.htm
- The Berkeley Electrical Engineering Annual Research Symposium (BEARS) featured an open house co-sponsored by Chess in order to display results for the benefit of our industrial partners and friends of the project. The program and presentations are available at http://www.eecs.berkeley.edu/BEARS/2007/index.html
- A weekly Chess workshop was held at Berkeley. The speakers and topics are listed in Section 4.4.3, and presentations for the workshop are available at http://chess.eecs.berkeley.edu/seminar.htm

We organize this section by thrust areas that we established in the statement of work.

2.1.1. Hybrid Systems Theory

We have proposed to build the theory of mixed discrete and continuous hybrid systems into a mathematical foundation of embedded software systems. For this purpose we have been pursuing four directions:

1. We have been designing models of computation that permit the composition of non-functional properties. Our work in composition of domain-specific modeling languages has supplemented the understanding we have gained in previous years on real-time and functional properties. We also continued our work on composition as a non-zero-sum game where the players (components) have different objectives.

2. We have been designing robust models of computation, where small perturbations of the system description cause only small changes in the system behavior. We have also been continuing our work in fault-tree analysis, where we generate threes based on dataflow graphs.

3. We have been developing and evaluating several methods for the computational treatment of hybrid systems. In particular, we have matured our design and implementation of a deterministic operational semantics for the simulation of hybrid systems, as well as ellipsoid-based algorithms for the efficient reach-set analysis of hybrid systems. Our work in game-based reasoning continues to inform how memoryless operations can be used to achieve some measure of optimality while still remaining within objectives of reachability.

4. We have been developing stochastic models that combine hybrid dynamics with sources of uncertainty. For controlling such stochastic systems, we improved the best known algorithms for solving stochastic games. We also pursued the application of stochastic hybrid models in systems biology and for other classes of small noise perturbation of deterministic hybrid systems.

2.1.1.a. Deep Compositionality

Compositional Specification of Behavioral Semantics
An emerging common trend in model-based design of embedded software and systems is the adoption of Domain-Specific Modeling Languages (DSMLs). While abstract syntax metamodeling enables the rapid and inexpensive development of DSMLs, the specification of DSML semantics is still a hard problem. In previous work, we have developed methods and tools for the semantic anchoring of DSMLs. Semantic anchoring introduces a set of reusable "semantic units" that provide reference semantics for basic behavioral categories using the Abstract State Machine (ASM) framework. In [21] we extend the semantic anchoring framework to heterogeneous behaviors by developing a method for the composition of semantic units. Semantic unit composition reduces the required effort from DSML designers and improves the quality of the specification.

Composing Different Models of Computation in Ptolemy II and Kepler
A model of computation (MoC) is a formal abstraction of execution in a computer. There is a need for composing MoCs in e-science. Kepler, which is based on Ptolemy II, is a scientific workflow environment that allows for MoC composition. This paper explains how MoCs are combined in Kepler and Ptolemy II and analyzes which combinations of MoCs are currently possible and useful. It demonstrates the approach by combining MoCs involving dataflow and finite state machines. The resulting classification should be relevant to other workflow environments wishing to combine multiple MoCs. Keywords: Model of computation, scientific workflow, Kepler, Ptolemy II [28].

Hybrid Geometric Reduction of Hybrid Systems
The reduction of mechanical systems with symmetries plays a fundamental role in understanding the many important and interesting properties of these systems. We have been working to generalize this result to a hybrid setting—a formidable obstacle to which is the copious mathematical framework needed to perform reduction. Such a generalization of reduction, therefore, requires a new method for viewing "hybrid objects," one that allows classical mathematical objects and morphisms between these objects to be easily "hybridized." The framework in which we have been working to carry out this generalization is that of category theory, and specifically through the notion of a hybrid object over a category (see [54]). Additional work is included in [76], where we show application to bipedal walkers. Techniques in Routhian reduction are presented in [82], and in [83] we discuss more uses for category theory to provide results in this area.

2.1.1.b. Robust Hybrid Systems

Fault Tree Analysis
In [70] we introduce a model-based approach to heterogeneous system design that enables the automatic generation of fault trees for analyzing system reliability properties. This approach extends our previous work that addressed the generation of fault trees from a dataflow model. In this new context, heterogeneous systems are composed of interacting discrete-time components, such as an electronic feedback controller, and continuous-time components, such as a plant. More recent work in computer-aided fault-tree generation methods is based on functional models of the system to produce a system fault tree automatically. Yet, most of these approaches were not applied to heterogeneous systems. Furthermore, these approaches continued to rely on intuition to create fault trees. Since in this approach fault tree generation is disjoint from the system modeling, consistency problems may arise when the structure and behavior of the system model is not accurately reflected. Our approach is different since we use a model of the system specified as a set of mathematical equations to derive the system fault modes and ultimately produce fault trees for heterogeneous systems.

2.1.1.c. Computational Hybrid Systems

Concurrent Reachability Games
A concurrent reachability game is a two-player game played on a graph: at each state, the players simultaneously and independently select moves; the two moves determine jointly a probability distribution over the successor states. The objective for player 1 consists in reaching a set of target states; the objective for player 2 is to prevent this, so that the game is zero-sum. Our contributions are two-fold. In [63], we discuss that for all epsilon>0, memoryless epsilon-optimal strategies exist. We also present a strategy-improvement (a.k.a. policy-iteration) algorithm for concurrent games with reachability objectives. Other work in [65] studies Nash Equilibria for memoryless strategies. This work, along with [67][62][63], addresses techniques and results for these applications.

Zeno Conditions
In [80], and in more detail in [91], we propose a technique to extend the simulation of a Zeno hybrid system beyond its Zeno time point. A Zeno hybrid system model is a hybrid system with an execution that takes an infinite number of discrete transitions during a finite time interval. We

argue that the presence of Zeno behavior indicates that the hybrid system model is incomplete by considering some classical Zeno models that incompletely describe the dynamics of the system being modeled. This motivates the systematic development of a method for completing hybrid system models through the introduction of new post-Zeno states, where the completed hybrid system transitions to these post-Zeno states at the Zeno time point. In practice, simulating a Zeno hybrid system is challenging in that simulation effectively halts near the Zeno time point. Moreover, due to unavoidable numerical errors, it is not practical to exactly simulate a Zeno hybrid system. Therefore, we propose a method for constructing approximations of Zeno models by leveraging the completed hybrid system model. Using these approximations, we can simulate a Zeno hybrid system model beyond its Zeno point and reveal the complete dynamics of the system being modeled.

Zeno systems are also discussed in [92], with respect to the stability properties of a class of Zeno equilibria.

2.1.1.d. Stochastic Hybrid Systems

Reachability analysis for controlled discrete time stochastic hybrid systems
A model for discrete time stochastic hybrid systems whose evolution can be influenced by some control input is proposed in this paper. With reference to the introduced class of systems, a methodology for probabilistic reachability analysis is developed that is relevant to safety verification. This methodology is based on the interpretation of the safety verification problem as an optimal control problem for a certain controlled Markov process. In particular, this allows characterizing through some optimal cost function the set of initial conditions for the system such that safety is guaranteed with sufficiently high probability. The proposed methodology is applied to the problem of regulating the average temperature in a room by a thermostat controlling a heater (see [52]).

Bounding Error for Stochastic Approximations
The work in [85] introduces, develops and discusses an integration-inspired methodology for the simulation and analysis of deterministic hybrid dynamical systems. When simulating hybrid systems, and thus unavoidably introducing some numerical error, a progressive tracking of this error can be exploited to discern the properties of the system, i.e., it can be used to introduce a stochastic approximation of the original hybrid system, the simulation of which would give a more complete representation of the possible trajectories of the system. Moreover, the error can be controlled to check and even guarantee (in certain special cases) the robustness of simulated hybrid trajectories.

2.1.2. Hybrid Systems – Components for Embedded Systems

2.1.2.a Building Efficient Simulations from Hybrid Bond Graph Models

Modern engineering systems are complex and made up of a large number of interacting components with nonlinear hybrid behaviors. This makes the building of accurate and computationally efficient simulation models a very challenging task. Recently, researchers and practitioners have adopted component- and actor-oriented frameworks for systematic construction of large models of complex systems. Such frameworks consist of mathematical

models for specifying individual component behavior and formal models of computation for defining component interactions. Simulation models are derived by representing the component behavior models as computational blocks, and the interaction models define the syntax and semantics of the connections between the blocks.

We have adopted the Hybrid Bond Graph (HBG) paradigm, an extension of the Bond Graph (BG) modeling language, for component-based modeling of embedded systems. It is a domain-independent topological modeling language that captures interactions among the different processes that make up the system and can be very effective in parameterized component-based modeling of hybrid systems. The challenge we face is in translating these models to computationally efficient simulation models.

The causal structure inherent in BG models provides the basis for conversion of BGs to efficient computational models. For HBGs, mode changes imply dynamic changes in the causal structure, and this alters the computational model during execution. We have developed a method for efficient simulation of HBG models by converting them to block diagram models, extending the procedure for BGs. Run-time changes are handled by reconfiguring the data flow paths within the blocks of the model. We demonstrate the technique by creating a computational model of an electrical power system in Matlab Simulink.

Our goal is to build efficient simulation models from HBG representations. The block diagram (BD) formalism is a widely used graphical, computational scheme for describing simulation models of continuous and hybrid systems (e.g., Ptolemy, Modelica, and Simulink, among others). We adopt the BD modeling paradigm, and develop a methodology for transforming HBGs to BDs [26][27].

We encounter two primary challenges in generating simulation models from HBG representations.

Challenge 1: Avoid pre-enumeration of model configurations. Consider a HBG model with m components and assume that each component has n_i switching junctions, where $i = 1, 2, \ldots m$. The HBG model, then, defines

$$\sum_{i=1}^{m} 2^{n_i}$$

n_i different system modes (or model configurations). When that number is large, it is infeasible to pre-enumerate all the model configurations before running the simulation. Therefore, mode changes, and reconfiguration of the BD model, have to be performed during run-time. Mode changes, implemented as junction switching, produce changes in the HBG model topology, which implies that the connections between blocks in the BD model may change dynamically during the simulations.

Challenge 2: Avoid algebraic loops. In component-based modeling, the underlying mathematical model is usually a set of differential-algebraic equations (DAEs). The DAE models may include algebraic loops. A system of equations with algebraic loops has a fixed point solution if the conditions for a unique solution are satisfied. However, generating the solution may become computationally expensive if the fixed point method needs many iterations to converge to a solution when algebraic loops are present. The order of equation evaluation, then, becomes very important.

When junction switches occur in a HBG model the following changes are made to the existing block diagram to generate the block diagram for the new mode.

1) Update the active HBG structure based on the junctions that change state. This procedure is described in Section 2.
2) Evaluate the changes in the determining bonds for the junctions in the HBG structure, and propagate these changes to derive the block diagram structure for the new mode.

For this work we implement the block diagram simulation models using Simulink. The Simulink environment provides all the primitives to implement the block diagram structure for a bond graph, and the bond graph elements. For hybrid junctions, we must implement the control structure as well as the dataflow structure. Rather than implementing a switching junction using discrete Simulink blocks, or using Stateflow extensions to Simulink, we implement the switching junction as custom written S-functions in C/C++. The S-function implements the dataflow machinery for the junction, as well as the evaluation of the control specification for the junction. For each bond connected to the junction, the S-function adds an input/output signal pair. The mapping of these signals to the effort/flow variables is determined dynamically. Note that we rely directly on the capabilities of the Simulink environment to detect the zero crossings, which define the mode changes. We have successfully tested this approach for developing a number of complex models for Advanced Life Support system applications.

In summary, our uses physical system modeling semantics as defined by BGs and HBGs to impose semantic structure on hybrid computational models in Simulink. Other elegant computational approaches, such as Ptolemy and HyVisual possess these semantics in a mathematical framework, but do not link these semantics to physical system principles. Therefore, we believe that our approach for building computational models from HBGs provides a comprehensive framework for starting from component-oriented physical system models and deriving efficient computational models for hybrid systems.

2.1.2.b Concurrency formalisms

Code Generation Frameworks

Embedded software requires concurrency formalisms other than threads and mutexes used in traditional programming languages like C. Actor-oriented design presents a high level abstraction for composing concurrent components. However, high level abstraction often introduces overhead and results in slower system. In [30], we address the problem of generating efficient implementation for the systems with such a high level description. We use partial evaluation as an optimized compilation technique for actor-oriented models. We use a helper-based mechanism, which results in flexible and extensible code generation framework. The end result is that the benefit offered by high level abstraction comes with (almost) no performance penalty. The code generation framework has been released in open source form as part of Ptolemy II.

Feedback in Strictly Causal Systems

In [46], we ask whether strictly causal components form well defined systems when arranged in feedback configurations. The standard interpretation for such configurations induces a fixed-point constraint on the function modeling the component involved. We bring together ideas from the areas of set theory, order theory, and theory of generalized ultrametric spaces to establish the existence of a unique fixed point for any such function constructively, what has resisted more traditional approaches and has been a much coveted albeit elusive goal.

2.1.2.c Application to system level design

System-level design (SLD) is considered by many as the next frontier in electronic design automation (EDA). SLD means many things to different people since there is no wide agreement on a definition of the term. Academia, designers, and EDA experts have taken different avenues to attack the problem, for the most part springing from the basis of traditional EDA and trying to raise the level of abstraction at which integrated circuit designs are captured, analyzed, and synthesized from. However, my opinion is that this is just the tip of the iceberg of a much bigger problem that is common to all system industry. In particular, I believe that notwithstanding the obvious differences in the vertical industrial segments (for example, consumer, automotive, computing, and communication); there is a common underlying basis that can be explored. This basis may yield a novel EDA industry and even a novel engineering field that could bring substantial productivity gains not only to the semiconductor industry but to all system industries including industrial and automotive, communication and computing, avionics and building automation, space and agriculture, and health and security, in short, a real technical renaissance (more information provided in [41]).

An Initial Study on Monetary Cost Evaluation for the Design of Automotive Electrical Architectures

One of the many challenges facing electronic system architects is how to provide a cost estimate related to design decisions over the entire life-cycle and product line of the architecture. Various cost modeling techniques may be used to perform this estimation. However, the estimation is often done in an ad-hoc manner, based on specific design scenarios or business assumptions. This situation may yield an unfair comparison of architectural alternatives due to the limited scope of the evaluation. A preferred estimation method would involve rigorous cost modeling based on architectural design cost drivers similar to those used in the manufacturing (e.g. process-based technical cost modeling) or in the enterprise software domain (e.g. COCOMO). This paper describes an initial study of a cost model associated with automotive electronic system architecture. The model's intended use is to evaluate system cost drivers in response to various architectural decisions (e.g. choosing a communication bus topology or mapping a function to hardware). The primary cost driver categories explored are design and development, part fabrication, assembly and in-service costs. The preliminary version of this cost model focuses on describing the key influences on cost, but not the entire mathematical model. The paper presents the cost model with the help of influence diagrams and illustrates the use of the cost modeling methodology through an automotive case study – a steer-by-wire system. As future work, we propose to build a cost model and supporting methodology that accounts for architecture evolution to address the issue of evolving architecture requirements as well as when and where to employ new technology in the architecture (see [32]).

Synthesis of task and message activation models in real-time distributed automotive systems

Modern automotive architectures support the execution of distributed safety- and time-critical functions on a complex networked system with several buses and tens of ECUs. Schedulability theory allows the analysis of the worst case end-to-end latencies and the evaluation of the possible architecture configurations options with respect to timing constraints. We present an optimization framework, based on an ILP formulation of the problem, to select the communication and synchronization model that exploits the trade-offs between the purely periodic and the precedence constrained data-driven activation models to meet the latency and

jitter requirements of the application. We demonstrated its effectiveness by optimizing complex real-life General Motors architecture (see [36]).

Optimizing end-to-end latencies by adaptation of the activation events in distributed automotive systems

Schedulability theory provides support for the analysis of the worst case latencies in distributed computations when the architecture of the system is known and the communication and synchronization mechanisms have been defined. In the design of complex automotive systems, however, a great benefit of schedulability analysis may come from its use as an aid in the exploration of the software architecture configurations that can best support the target application. We present an optimization algorithm that leverages the trade-offs between the purely periodic and the data-driven activation models to meet the latency requirements of distributed vehicle functions. We demonstrate its effectiveness on a complex automotive architecture (see [37]).

2.1.3. Model-based Design

Model-based design of embedded systems is predicated on the notion that models play an essential role in the entire life-cycle of systems from specification, design, development, verification, integration to upgrade and maintenance. Our approach refines the overall concept of "model-driven development" (advocated for example by the Model-Driven Architecture (MDA) of the Object Management Group, www.omg.org/mda) by emphasizing two novel elements: (1) the use of domain-specific modeling languages, and (2) the integration of formal analysis tools, verification techniques and model transformations in the development process.

Domain-specific modeling languages (DSMLs). Our approach to the development of embedded systems is centered on analyzable models that capture the developer's design intent. Domain specificity means that modeling languages are tailored to the needs of the application domain. The success of this approach has been demonstrated by existing tools like Simulink/Stateflow and Matrix-X, which are widely used in the aerospace/automotive industry for flight control/vetronics software development. In spite of their successes, these tools exhibit two serious shortcomings: (1) It is impossible to reason formally about and verify the code generated by the tools since the semantics of their modeling languages are not precisely defined, and (2) It is very hard to integrate the generated code into a larger system which contains parts that use other models of computation.

We have continued our research on a technology that supports the precise definition of the abstract syntax and well-formedness rules for DSMLs through the use of meta-modeling and meta-programmable tools [11]. We have also extended our model transformation tool suite and deepened its theoretical foundation.

2.1.3.a Composition of Domain Specific Modeling Languages

Model-based design frameworks that aggressively use DSMLs need to support the composition of modeling languages. For example, the MIC infrastructure uses abstract syntax metamodeling and meta-programmable tool suites for the rapid construction of DSMLs with well defined syntax and semantics. We focused our efforts on the semantic foundation of modeling languages and their extension to sensor network application domain.

During the last year we achieved the following progress in this research:

1. We have developed new theory for defining the structural semantics of domain specific modeling language and developed tools for composing and comparing metamodels [12][13][14][15]. Despite the fundamental role of structure in the model-based approach, it remained largely unformalized. To address this and other issues, we developed a mathematical formulation of structure, giving a precise tool-independent definition that is amendable to formal analysis. This work also provides a formal understanding of model transformations and metamodeling, yielding a complete picture of the structural basis of model-based design. In this sense, we broaden the term structural semantics to include the semantics of model transformation and metamodeling.

2. We have developed the second release of a semantic anchoring tool suite [19][25] that comprises (1) the ASM-based AsmL tool suite from Microsoft Research for specifying semantic units and (2) the MIC modeling (GME) and model transformation (GReAT) tool suites that support the specification of transformation between the DSML metamodels and the Abstract Data Models used in the semantic units. Our attention turned to the formulation of semantic units (as opposed to the technique for their formal specification). We continued experimenting with building alternative derivations for "systems of semantic units" that logically bring together behavioral and interaction categories. This work closely related to Berkeley's work on abstract semantics but approaches the problem from a very different angle.

3. We have continued developing the foundations for the compositional specification of behavioral semantics in the semantic anchoring framework [17][18][21]. In the semantic anchoring infrastructure, we define a finite set of semantic units, which capture the semantics of basic behavioral and interaction categories. If the semantics of a DSML can be directly anchored to one of these basic categories, its semantics can be defined by simply specifying the model transformation rules between the metamodel of the DSML and the Abstract Data Model of the semantic unit [18]. However, in heterogeneous systems, the semantics is not always fully captured by a predefined semantic unit. If the semantics is specified from scratch (which is the typical solution if it is done at all) it is not only expensive but we loose the advantages of anchoring the semantics to (a set of) common and well-established semantic units. This is not only loosing reusability of previous efforts, but has negative consequences on our ability to relate semantics of DSMLs to each other and to guide language designers to use well understood and safe behavioral and interaction semantic "building blocks" as well.
   Our proposed solution is to define semantics for heterogeneous DSMLs is the composition of semantic units. If the composed semantics specifies a behavior which is frequently used in system design (for example, the composition of SDF interaction semantics with FSM behavioral semantics defines semantics for modeling signal processing systems), the resulting semantics can be considered a *derived semantic unit*, which is built on *primary semantic units*, and could be offered up as one of the set of semantic units for future anchoring efforts. Note that *primary semantic units* refer to the semantic units that capture the semantics of the *basic behavioral categories*, such as FSM, TA and HA. The composition approach we describe in the rest of the paper is

strongly influenced by Gossler and Sifakis framework for composition by clearly separating behavior and interaction.

4. We continued our efforts in migrating research results into stable tool suites and presenting their use for the broader community [11][23][24]. Some of our complex applications turned our attention to the challenges of managing large models. This requires the development of a variety of techniques for the decomposition and recomposition of large model databases. The unique challenge is the reconstruction of model consistency after model recomposition.
5. We have extended model-based design approaches to developing sensor network applications on SOA platforms [8][10][22].

2.1.3.b Model Transformations

In the model-based design area, we have further developed and refined our model transformation tool suite. Based on practical feedback from users who have used the model transformation language GReAT, we have developed a suite of new capabilities, listed below. The updated language and tools are available from the ESCHER website (http://escher.isis.vanderbilt.edu).

- Templatized transformation rules. This is a construct, similar to the 'template functions' of C++ or 'higher-order functions' of ML and Haskell, where the type of specific rule elements is not specified at rule definition time, rather when the rule is used in a context. 'Template rules' are almost identical to normal rules, except the type of pattern variables (i.e. classes) is not bound. When a 'template rule' is used in a larger model transformation, then the designer has the opportunity to 'bind' the type of the pattern variable to a concrete type. This improves readability and the reusability of transformation rules.
- 'Blockification' tool. When constructing complex transformations, we have observed that it is a very common operation when the user groups together a certain sequence of rewriting rules into a block which is then re-used on a higher level. When done manually, this involves a lot of tedious editing operations that are error-prone. We have developed an interactive tool that allows selecting a group of rules and forms a higher-level block from them that is inserted into their place. This tool is a 'modeler's aid' for constructing complex transformations. The tool checks several consistency properties and executes the 'blockification' only if the result does not violate the semantics of the original transformation program.
- 'Next rule connector' tool. Similarly to the tool explained above, this tool supports the rapid, sequential 'wiring' of two transformation rules. The designer has to select an 'upstream' and 'downstream' rule, and the tool connects them up using the ordering of the ports of the rules. The tool performs a semantic check on the compatibility between the rules, and establishes the connections only if they are legal.
- 'Group' operator. One common operation in transforming designs (using graph transformation techniques) is the deletion, copying, or moving entire subgraphs. While this could be done with elementary operations of node/edge addition and removal, their modeling is rather cumbersome. To support such operations in a convenient manner, we have introduced a new type of rewriting rule into the transformation language: the 'group' rule. The group rule matches the host graph and generates (potentially multiple) matches for the pattern nodes and edges. Now, the designer can select specific pattern

nodes and edges to form ‘groups’ (i.e. sub graphs) from the matched elements, which are treated as a unit, a ‘cluster’. These clusters can then be deleted/copied/moved as a unit in the action part of the rule. The detailed design of this operator has been published in [1]. We have found a number of application examples for the new operator; these have been published in [2]. Both the GREAT interpreter and the code generator support now this new operator.

We have used the GReAT tool suite during the year in developing model transformations for this ITR project, as well as for other sponsored research projects. In the context of this ITR, we have used GReAT to develop a model transformer that uses a transformational approach to ‘platform modeling’. In another, NASA-sponsored project we have used the language to develop a Simulink and Stateflow code generator that compiles (restricted) models into C code and another, intermediate language that can then be subjected to static code analysis to prove safety-critical properties of the code (e.g. no buffer overflows). During the year, we have continued making improvements to the GReAT and UDM packages.

Another related effort focused on platform modeling and the connection between modeling languages and analysis tools.

In the platform modeling area, we developed a new “platform modeling language” (PML) that allows the rapid construction of “design model -> analysis model” transformation tools. In this language, one has to explicitly include an extended metamodel for the platform that is assumed by the design modeling language. This extended metamodel captures how the “components” and the “run-time kernel” of the target run-time system look like in terms of concepts of language of the analysis tool. Additionally, the language supports a high-level specification of the mapping of design language structures into the “components” and the “run-time kernel”. This mapping is specified using a higher-order, graph-transformation-based language that is simpler than GReAT. The results of this work have been reported in [4] and [6] , the latter being a PhD thesis.

One of the most difficult problems in model transformations is to ensure their correctness. For complex, model-based design tool chains, it is essential that the model transformations connecting the different tools preserve semantics of the models they translate. We have started working on this problem and one approach, based on verifying the correctness of specific instances of transformations has been developed for specific set of examples. Early results have been published in [5] and [7]. These techniques can be considered as light-weight formal methods for producing an ‘independent certificate’ for each execution of the transformation that proves that the models transformed preserve some property of interest (e.g. reachability in finite transition systems).

2.1.4 Experimental Research

The main emphasis of our research is on the foundations of hybrid systems theory and of embedded system design. However, in the best tradition of our groups, a strong application program is necessary to verify the viability of the theory and to uncover difficult problems that provide appropriate motivation to develop new methods and theories. Most of the applications studied are distributed systems where scarce and fragile resources have to be used to provide reliable behavior. Wireless sensor networks, distributed systems for automotive electronics, embedded systems for national and homeland defense, are but a few examples that attracted the attention of our research groups because of their complexity and of their objective importance.

We argue that the distributed nature of the applications poses additional challenges to overcome with an appropriate design methodology and supporting tools.

In particular, during this period, we have focused on the application to UAVs for air borne combat, Unmanned Underwater Vehicles, wireless sensor networks to control and monitoring, on fault-diagnosis, fault-adaptive and fault-tolerant approaches for distributed systems, and finally, on a multi-media problem as a test vehicle for the methodology and the tools embedded in Metropolis.

2.1.4.a. Embedded Control Systems

Decentralized control of unmanned underwater vehicles
A decentralized control scheme for large packs of unmanned underwater vehicles (UUV) using reachability computation and model predictive control is proposed and investigated. This scheme fits within a broader template based method for organizing and controlling large groups of UUVs and addressed one important mode of operation for these vehicles, specifically decentralized, coordinated group pursuit and tracking of intruders through the UUV network. The reachability computation and model predictive control schemes are presented along with simulation results with medium-sized packs of UUVs. Hardware implementation is also described for five UUVs (see [31]).

2.1.4.b. Embedded Software for National and Homeland Security

Autonomous Ground Vehicles
Ground vehicle research, although more stable than airborne vehicles, presents the subtle problem of providing intelligent behavior which operates in real-time, executes safely, and yet provides a "smooth" reaction to stimuli—i.e., the software behavior is somewhat humanized. Our application is the DARPA Urban Challenge, and we are using the ground vehicle testbed to show the performance of various algorithms and advancements in theory of switched systems, model-predictive control, parameter identification, time-triggered distributed components, and computer vision, as well as how these components work in real-time with one another. Additional concerns which this project addresses are distributed software testing, component-based design, and vehicle/sensor health monitoring. We are proving many of the theories and algorithms which have been developed in the last four years of the ITR.

Real-time Computer Vision
The goal of this task is to detect moving objects using a stereo camera system mounted on a car. This information will be used to detect, and estimate the trajectories of (possibly) mobile obstacles such as other vehicles. Note that, in this scenario, we wish to detect both objects in close proximity to our vehicle (for example, the vehicle in front of us), and also objects that are farther away (for example, oncoming vehicles), so we cannot make assumptions about whether height errors at adjacent pixels appear to be close to planar. Our solution must also be able to run in real time. In this application, we are interested in determining 3D motion in the scene, rather than the more-studied case of 2D motion in the image. In particular, we want to find the motion of objects in the scene relative to each other; if we have vehicle state data available, or if we can assume that the majority of the scene is not moving, we can determine which motion-segmented

region corresponds to the background and remove its perceived motion from the perceived motion of the other objects, obtaining the motion of the other objects in a global frame.

2.1.4.c. Networks of Distributed Sensors

VisualSense: Visual Editor and Simulator for Wireless Sensor Network Systems
VisualSense is a modeling and simulation framework for wireless and sensor networks that builds on and leverages Ptolemy II. Modeling of wireless networks requires sophisticated representation and analysis of communication channels, sensors, ad-hoc networking protocols, localization strategies, media access control protocols, energy consumption in sensor nodes, etc. This modeling framework is designed to support a component-based construction of such models. It supports actor-oriented definition of network nodes; wireless communication channels, physical media such as acoustic channels, and wired subsystems. The software architecture consists of a set of base classes for defining channels and sensor nodes, a library of subclasses that provide certain specific channel models and node models, and an extensible visualization framework. Custom nodes can be defined by subclassing the base classes and defining the behavior in Java or by creating composite models using any of several Ptolemy II modeling environments. Custom channels can be defined by subclassing the WirelessChannel base class and by attaching functionality defined in Ptolemy II models. It is intended to enable the research community to share models of disjoint aspects of the sensor nets problem and to build models that include sophisticated elements from several aspects. VisualSense can be downloaded from http://ptolemy.eecs.berkeley.edu/visualsense/.

Viptos: a Programming Models for Sensor Networks
Viptos (Visual Ptolemy and TinyOS) is an integrated graphical development and simulation environment for TinyOS-based wireless sensor networks. Viptos allows developers to create block and arrow diagrams to construct TinyOS programs from any standard library of nesC/TinyOS components. The tool automatically transforms the diagram into a nesC program that can be compiled and downloaded from within the graphical environment onto any TinyOS-supported target hardware. In particular, Viptos includes the full capabilities of VisualSense, which can model communication channels, networks, and non-TinyOS nodes. Viptos is compatible with nesC 1.2 and includes tools to harvest existing TinyOS components and applications and convert them into a format that can be displayed as block (and arrow) diagrams and simulated.

Viptos is based on TOSSIM and Ptolemy II. TOSSIM is an interrupt-level simulator for TinyOS programs. It runs actual TinyOS code but provides software replacements for the simulated hardware and models network interaction at the bit or packet level. Ptolemy II is a graphical software system for modeling, simulation, and design of concurrent, real-time, embedded systems. Ptolemy II focuses on assembly of concurrent components with well-defined models of computation that govern the interaction between components. VisualSense is a Ptolemy II environment for modeling and simulation of wireless sensor networks at the network level.

Viptos provides a bridge between VisualSense and TOSSIM by providing interrupt-level simulation of actual TinyOS programs, with packet-level simulation of the network, while allowing the developer to use other models of computation available in Ptolemy II for modeling various parts of the system. While TOSSIM only allows simulation of homogeneous networks where each node runs the same program, Viptos supports simulation of heterogeneous networks

where each node may run a different program. Viptos simulations may also include non-TinyOS-based wireless nodes. The developer can easily switch to different channel models and change other parts of the simulated environment, such as creating models to generate simulated traffic on the wireless network.

Viptos inherits the actor-oriented modeling environment of Ptolemy II, which allows the developer to use different models of computation at each level of simulation. At the lowest level, Viptos uses the discrete-event scheduler of TOSSIM to model the interaction between the CPU and TinyOS code that runs on it. At the next highest level, Viptos uses the discrete-event scheduler of Ptolemy II to model interaction with mote hardware, such as the radio and sensors. This level is then embedded within VisualSense to allow modeling of the wireless channels to simulate packet loss, corruption, delay, etc. The user can also model and simulate other aspects of the physical environment including those detected by the sensors (e.g., light, temperature, etc.), terrain, etc. Viptos can be downloaded from http://ptolemy.eecs.berkeley.edu/viptos/.

Control of Communication Networks

In a series of papers Abate and co-authors have continued to explore using stochastic hybrid systems congestion control schemes for both wired and wireless networks. These methods have tremendous applicability to other classes of network embedded systems as well (see [77][86]).

2.2. Project Findings

Abstracts for key publications representing project findings during this reporting period, are provided here. A complete list of publications that appeared in print during this reporting period is given in Section 4 below, including publications representing findings that were reported in the previous annual report.

[1] D. Balasubramanian, A. Narayanan, S. Neema, F. Shi, R. Thibodeaux, G. Karsai: A Subgraph Operator for Graph Transformation Languages, Proceedings of 6th International Workshop on Graph Transformation and Visual Modeling Techniques, Braga, Portugal.
In practical applications of graph transformation techniques to model transformations one often has the need for copying, deleting, or moving entire subgraphs that match a certain graph pattern. While this can be done using elementary node and edge operations, the transformation is rather cumbersome to write. To simplify the transformation, we have recently developed a novel approach that allows selecting subgraphs from the matched portion of the host graph, applying a filter condition to the selection, and performing a delete, move, or copy operation on the filtered result in the context of a transformation rule. The approach has been implemented in the GReAT language and tested on examples that show the practical efficacy of the technique. The paper describes the technique in detail and illustrates its use on a real-life example.

[2] D. Balasubramanian, A. Narayanan, S. Neema, B. Ness, F. Shi, R. Thibodeaux, and G. Karsai: "Applying a Grouping Operator in Model Transformations", submitted to 3rd International Workshop on Graph and Model Transformation (GraMoT). (Submitted.)
The usability of model transformation languages depends on the level of abstractions one can work with in rules to perform complex operations on models. Recently, we have

introduced a novel operator for our model transformation language GReAT that allows the concise specification of complex model (graph) rewriting operations. In this paper we show how the new operator can be used to implement non-trivial model manipulations with fewer and simpler rules, while maintaining efficiency. The examples were motivated by problems encountered in real-life model transformations.

[3] A. Agrawal, Gabor Karsai, Sandeep Neema, Feng Shi, Attila Vizhanyo: "The Design of a Language for Model Transformations", Journal on Software and System Modeling, pp 261-288, Volume 5, Number 3 / September, 2006.
Model-driven development of software systems envisions transformations applied in various stages of the development process. Similarly, the use of domain-specific languages also necessitates transformations that map domain-specific constructs into the constructs of an underlying programming language. Thus, in these cases, the writing of transformation tools becomes a first-class activity of the software engineer. This paper introduces a language that was designed to support implementing highly efficient transformation programs that perform model-to-model or model-to-code translations. The language uses the concepts of graph transformations and metamodeling, and is supported by a suite of tools that allow the rapid prototyping and realization of transformation tools.

[4] T. Szemethy, Gabor Karsai: "PML: a Language for Platform Modeling," Electronic Communications of the EASST, Volume 4, 2006: Graph and Model Transformation 2006.
Modeling the computational platforms is necessary to analyze the execution characteristics of systems developed using a model-based approach. In this paper, we introduce a novel platform modeling language: PML that is based on (a) transformational concepts borrowed from graph transformation languages, and (b) generative concepts from platform modeling, like 'kernel skeleton'. PML relies on higher-level, compact constructs that represent a special case of model transformations, and which are then used to specify platform semantics. The paper also illustrates how PML constructs can be compiled into lower-level constructs of more traditional model transformational languages, such as GReAT.

[5] A. Narayanan, Gabor Karsai: "Using Semantic Anchoring to Verify Behavior Preservation in Graph Transformations," Electronic Communications of the EASST, Volume 4, 2006: Graph and Model Transformation 2006.
Graph transformation is often used to transform domain models from one domain specific language (DSML) to another. In some cases, the DSMLs are based on a formalism that has many implementation variants, such as Statecharts. For instance, it could be necessary to transform iLogix Statechart models into Matlab Stateflow models. The preservation of behavior of the models is crucial in such transformations. Bisimulation has previously been demonstrated as an approach to verifying behavior preservation, and semantic anchoring is an approach to specifying the dynamic semantics of DSMLs. We propose a method to verify behavior preservation, using bisimulation in conjunction with semantic anchoring. We will consider two hypothetical variants of the Statecharts formalism, and specify the operational semantics of each variant by semantic anchoring, using Abstract State Machines as a common semantic framework. We then establish

bisimulation properties to verify if the behavior models of the source and target Statechart models are equivalent for a particular execution of the transformation.

[6] T. Szemethy: Domain-Specific Models, Model Analysis, Model Transformations. PhD thesis, Vanderbilt University, 2006.
This dissertation proposes a novel approach, applicable in the design-time analysis and verification of computer-based systems. The proposed approach, platform modeling, constructs analysis models capturing the system's behavior on a particular implementation platform. The approach is discussed in the context of Model-Integrated Computing, which is a development methodology leveraging on the use of domain-specific modeling languages and advanced model transformation techniques. During development, platform-independent design models are refined into platform-specific models using model transformation. The main contribution of this work is the enhancement of this process with the automatic generation of analysis models. These analysis models assign platform-specific semantics to the design model. This assignment is done through a transformational approach, using graph transformation. The dissertation presents a case study using a conventional graph transformation tool. Then, a new graph transformation language designed specifically for such transformations is proposed, and its advantages are demonstrated.

[7] G. Karsai, A. Narayanan: On the Correctness of Model Transformations in the Development of Embedded Systems, Proceedings of the 2006 Monterey Workshop (submitted), Paris, France, Oct. 2006.
Model based techniques have become very popular in the development of software for embedded systems, with a variety of tools for design, simulation and analysis of model based systems being available (such as Matlab's Simulink, the model checking tool NuSMV, etc.). Model transformations usually play a critical role in such model based development approaches. While the available tools are geared to verify properties about individual models, the correctness of model transformations is generally not verified. However, errors in the transformation could present serious problems. Proving a property for a certain source model becomes irrelevant if an erroneous transformation produces an incorrect target model. One way to provide assurance about a transformation would be to prove that it preserves certain properties of the source model (such as reachability) in the target model. In this paper, we present some general approaches to providing such assurances about model transformations. We will present some case studies where these techniques can be applied.

[8] Isaac Amundson, Manish Kushwaha, Xenofon Koutsoukos, Sandeep Neema, and Janos Sztipanovits. "OASiS: A Service-Oriented Middleware for Pervasive Ambient-Aware Sensor Networks", *Pervasive and Mobile Computing Journal on Middleware for Pervasive Computing.* (Submitted October 2006.)
Heterogeneous sensor networks consisting of networked devices embedded into the physical world have a significant role in pervasive computing systems. Such sensor networks may contain wireless sensor networks that are ensembles of small, smart, and cheap sensing and computing devices that permeate the environment, as well as high-bandwidth rich sensors such as satellite imaging systems, meteorological stations, air

quality stations, and security cameras. Emergency response, homeland security, and many other applications have a very real need to interconnect such diverse networks and access information in real-time. While Web service standards provide well-developed mechanisms for resource-intensive computing nodes, linking such mechanisms with wireless sensor networks is very challenging because of limited resources, volatile communication links, and often node mobility. This paper presents a service-oriented programming model and middleware for ad-hoc wireless sensor networks which permits discovery and access of Web services. Sensor network applications are realized as graphs of modular and autonomous services with well-defined interfaces that allow them to be published, discovered, and invoked over the network, providing a convenient mechanism for integrating services from heterogeneous sensor systems. Our approach provides dynamic discovery, composition, and binding of services based on an efficient localized constraint satisfaction algorithm that can be used for developing ambient-aware applications that adapt to changes in the environment. A tracking application that employs many inexpensive sensor nodes, as well as a Web service, is used to illustrate the approach. Our results demonstrate the feasibility of ambient-aware applications that interconnect wireless sensor networks and Web services.

[9] Matthew Emerson and Sandeep Neema and Janos Sztipanovits: "Metamodeling Languages and Metaprogrammable Tools," in *Handbook of Real-Time and Embedded Systems*, Ed. Insup Lee, Joseph Leung, Sang H. Son, CRC Press, 2006
Model-Integrated Computing, one practical manifestation of OMG's Model Driven Architecture, advocates the development of domain specific modeling languages for system design, specification, and analysis. Recent years have seen a variety of new tools and metamodeling languages which support MIC-style model-based design, including the Eclipse Modeling Framework (EMF) and the Microsoft Domain-Specific Languages (DSL) tools. This situation introduces new challenges for system designers, including the challenge of deciding which tool and language to adopt for a particular project. This paper provides two contributions to the research on model-based design. First, we provide a technical comparison of three modeling languages and their supporting tools: GME's MetaGME, EMF's Ecore, and Microsoft's developing Domain Model Designer language. Through this comparison we show that the consequences of choosing one metamodeling language over another are non-critical for language specification, since all of the metamodeling languages support the same fundamental language design concepts. Second, we generalize previous work to outline a set of tools and procedures which may be used to adapt an existing metaprogrammable tool suite to support new metamodeling languages while maintaining backwards compatibility with existing tools and models.

[10] Isaac Amundson, Manish Kushwaha, Xenofon Koutsoukos, Sandeep Neema, and Janos Sztipanovits. "Efficient Integration of Web Services in Ambient-aware Sensor Network Applications". *3rd IEEE/CreateNet International Workshop on Broadband Advanced Sensor Networks (BaseNets 2006)*, San Jose, CA, Oct. 2006.
Sensor webs are heterogeneous collections of sensor devices that collect information and interact with the environment. They consist of wireless sensor networks that are ensembles of small, smart, and cheap sensing and computing devices that permeate the environment as well as high-bandwidth rich sensors such as satellite imaging systems,

meteorological stations, air quality stations, and security cameras. Emergency response, homeland security, and many other applications have a very real need to interconnect such diverse networks and access information in real-time. While Internet protocols and Web standards provide well-developed mechanisms for accessing this information, linking such mechanisms with resource-constrained sensor networks is very challenging because of the volatility of the communication links. This paper presents a service-oriented programming model for sensor networks which permits discovery and access of Web services. Sensor network applications are realized as graphs of modular and autonomous services with well-defined interfaces that allow them to be described, published, discovered, and invoked over the network providing a convenient way for integrating services from heterogeneous sensor systems. Our approach provides dynamic discovery, composition, and binding of services based on an efficient localized constraint satisfaction algorithm that can be used for developing ambient-aware applications that adapt to changes in the environment. A tracking application that employs many inexpensive sensor nodes, as well as a Web service, is used to illustrate the approach. Our results demonstrate the feasibility of ambient-aware applications that interconnect wireless sensor networks and Web services.

[11] Karsai, G., Ledeczi, A., Neema, S., Sztipanovits, J.: The Model-Integrated Computing Toolsuite: Metaprogrammable Tools for Embedded Control System Design, *Proc. of the IEEE Joint Conference CCA, ISIC and CACSD*, Munich, Germany, 2006. Model-Integrated Computing is a development approach that advocates the use of Domain-Specific Modeling throughout the system development process and lifecycle. This paper describes and summarizes the generic and reusable soft-ware tools that support MIC and which can be tailored to solve a wide variety of modeling, analysis, and generation problems in an engineering process.

[12] Jackson, E., Sztipanovits, J.: "Towards A Formal Foundation For Domain Specific Modeling Languages," *Proceedings of the Sixth ACM International Conference on Embedded Software (EMSOFT'05)*, pp. 53-63, Seoul, Korea October 22-25, 2006 Embedded system design is inherently domain specific and typically model driven. As a result, design methodologies like OMG's model driven architecture (MDA) and model integrated computing (MIC) evolved to support domain specific modeling languages (DSMLs). The success of the DSML approach has encouraged work on the heterogeneous com- position of DSMLs, model transformations between DSMLs, approximations of formal properties within DSMLs, and reuse of DSML semantics. However, in the effort to produce a mature design approach that can handle both the structural and behavioral semantics of embedded system design, many foundational issues concerning DSMLs have been overlooked. In this paper we present a formal foundation for DSMLs and for their construction within metamodeling frameworks. This foundation allows us to algorithmically decide if two DSMLs or metamodels are equivalent, if model transformations preserve properties, and if metamodeling frameworks have meta-metamodels. These results are key to building correct embedded systems with DSMLs.

[13] Jackson, E., Sztipanovits, J.: "Constructive Techniques for Meta and Model

Level Reasoning," Models 07, (submitted)
The structural semantics of UML-based metamodeling were recently explored [1], providing a characterization of the models adhering to a metamodel. In particular, metamodels can be converted to a set of constraints expressed in a decidable subset of first-order logic, an extended Horn logic. We augment the constructive techniques found in logic programming, which are also based on an extended Horn logic, to produce constructive techniques for reasoning about models and metamodels. These methods have a number of practical applications: At the meta-level, it can be decided if a (composite) metamodel characterizes a non-empty set of models, and a member can be automatically constructed. At the model-level, it can be decided if a submodel has an embeddeding in a well-formed model, and the larger model can be constructed. This amounts to automatic model construction from an incomplete model. We describe the concrete algorithms for constructively solving these problems, and provide concrete examples.

[14] Jackson, E., Sztipanovits, J.: "Models as Structures: The Structural Semantics of Model-Based Design," Journal of Software and Systems Modeling (SOSYM), (submitted)
Model-based approaches to system design are now widespread and successful. These approaches make extensive use of model structure to facilitate domain specific abstractions and platform modeling. However, the structural semantics of model-based approaches are not well-understood. In this paper we develop the formal foundations for the structural semantics of model-based design. Additionally, we show how our formalization can be applied to existing tools, and how it yields algorithms for the analysis of domain-specific modeling languages (DSMLs) and model transformations.

[15] Emerson M., Sztipanovits J.: "Techniques for Metamodel Composition", *OOPSLA – 6th Workshop on Domain Specific Modeling*, 123-139, Portland, Oregon, October 22, 2006
The process of specifying an embedded system involves capturing complex interrelationships between the hardware domain, the software domain, and the engineering domain used to describe the environment in which the system will be embedded. Developers increasingly turn to domain-specific modeling techniques to manage this complexity, through such approaches as Model Integrated Computing and Model Driven Architecture. However, the specification of domain-specific modeling language syntax and semantics remains more of an art than a science. Typically, the syntax of a DSML is captured using a metamodel; however, there are few best-practices for metamodeling and no public collection of reusable metamodel to address common language specification requirements. There is a need for an advanced, comprehensive language design environment that offers tool support for a wide range of metamodel reuse strategies and the preservation of metamodeling best-practices. We outline existing techniques for the reuse and composition of metamodels, and propose a new metamodel composition technique we call Template Instantiation.

[16] Emerson M., Duncavage S., Mathe J., Sztipanovits J.: "WiNeSim: A Wireless Network Simulation Tool", *EMSOFT – 1rst International Workshop on Embedded*

Systems Security, Seoul, South Korea, October 26, 2006
We provide an overview of WiNeSim, a highly extensible wireless network modeling and simulation tool with a network attack modeling component. WiNeSim provides a high-level graphical modeling interface for the rapid declarative specification of network configurations, including the selection of node hardware, MAC protocols, and routing protocols. Furthermore, it enables users to specify certain network nodes to act as "smart" attackers who adapt their behavior and attack styles based on perceived network conditions in accordance with user-specified algorithms given as timed automata. WiNeSim is designed to easily integrate new network protocols and attack styles.

[17] Sztipanovits, J.: "Towards the Compositional Specification of Semantics for Heterogeneous Domain-Specific Modeling Languages," *Workshop on Foundations and Applications of Component-based Design*. Seoul, Korea October 26, 2006
Domain-Specific Modeling Languages (DSMLs) play fundamental role in the model-based design of embedded software and systems. While abstract syntax metamodeling enables the rapid and inexpensive development of DSMLs, the specification of DSML semantics is still a hard problem. In previous work, we have developed methods and tools for the semantic anchoring of DSML-s. Semantic anchoring introduces a set of reusable "semantic units" that provides reference semantics for basic behavioral categories using the Abstract State Machine framework. In this paper, we extend the semantic anchoring framework to heterogeneous behaviors by developing method for the composition of semantic units. Semantic unit composition reduces the required effort from DSML designers and improves the quality of the specification. The proposed method is demonstrated through a case study.

[18] Kai Chen, Janos Sztipanovits, Sandeep Neema: "Compositional Specification of Behavioral Semantics," *Technical report, ISIS-06-705,* 2006.
Domain-Specific Modeling Languages (DSMLs) play fundamental role in the model-based design of embedded software and systems. While abstract syntax metamodeling enables the rapid and inexpensive development of DSMLs, the specification of DSML semantics is still a hard problem. In previous work, we have developed methods and tools for the semantic anchoring of DSMLs. Semantic anchoring introduces a set of reusable "semantic units" that provide reference semantics for basic behavioral categories using the Abstract State Machine framework. In this paper, we extend the semantic anchoring framework to heterogeneous behaviors by developing method for the composition of semantic units. Semantic unit composition reduces the required effort from DSML designers and improves the quality of the specification. The proposed method is demonstrated through a case study.

[19] Kai Chen, Janos Sztipanovits, Sandeep Neema: "A Case Study on Semantic Unit Composition," Workshop on Modeling in Software Engineering, ICSE 2007 (MISE 2007).
In previous work we have discussed a semantic anchoring framework that enables the semantic specification of Domain-Specific Modeling Languages by specifying semantic anchoring rules to predefined semantic units. This framework is further extended to support heterogeneous systems by developing a method for the composition of semantic

units. In this paper, we explain the semantic unit composition through a case study.

[20] Sztipanovits, J., Bay, J., Rohrbough,, L., Sastry, S., Schmidt, D., Whitaker, N., Winter, D.: "ESCHER: A New Technology Transitioning Model in Embedded Systems and Software," *IEEE Computer*, March, 2007
The paper describes a new transitioning model, ESCHER, that focuses on establishing a quality controlled repository supported by industry.

[21] Kai Chen, Janos Sztipanovits, and Sandeep Neema: "Compositional Specification of Behavioral Semantics," *Proceedings of Design Automation and Test in Europe Conference (DATE 07)*, Nice, France, April 2-5, 2007
An emerging common trend in model-based design of embedded software and systems is the adoption of Domain-Specific Modeling Languages (DSMLs). While abstract syntax metamodeling enables the rapid and inexpensive development of DSMLs, the specification of DSML semantics is still a hard problem. In previous work, we have developed methods and tools for the semantic anchoring of DSMLs. Semantic anchoring introduces a set of reusable "semantic units" that provide reference semantics for basic behavioral categories using the Abstract State Machine (ASM) framework. In this paper, we extend the semantic anchoring framework to heterogeneous behaviors by developing a method for the composition of semantic units. Semantic unit composition reduces the required effort from DSML designers and improves the quality of the specification. The proposed method is demonstrated through a case study.

[22] Manish Kushwaha, Isaac Amundson, Xenofon Koutsoukos, Sandeep Neema, and Janos Sztipanovits. "OASiS: A Programming Framework for Service-Oriented Sensor Networks". *In IEEE/Create-Net COMSWARE 2007*, Bangalore, India, Jan. 2007
Wireless sensor networks consist of small, inexpensive devices which interact with the environment, communicate with each other, and perform distributed computations in order to monitor spatio-temporal phenomena. These devices are ideally suited for a variety of applications including object tracking, environmental monitoring, and homeland security. At present, sensor network technologies do not provide off-the-shelf solutions to users who lack low-level network programming experience. Because of limited resources, ad hoc deployments, and volatile wireless communication links, the development of distributed applications require the combination of both application and system-level logic. Programming frameworks and middleware for traditional distributed computing are not suitable for many of these problems due to the resource constraints and interactions with the physical world. To address these challenges we have developed OASiS1, a programming framework which provides abstractions for object-centric, ambient-aware, service-oriented sensor network applications. OASiS uses a well-defined model of computation based on globally asynchronous locally synchronous dataflow, and is complemented by a user-friendly modeling environment. Applications are realized as graphs of modular services and executed in response to the detection of physical phenomena. We have also implemented a suite of middleware services that support OASiS to provide a layer of abstraction shielding the low-level system complexities. A tracking application is used to illustrate the features of OASiS. Our results demonstrate the feasibility and the benefits of a service-oriented programming framework for

composing and deploying applications in resource constrained sensor networks.

[23] Fabrice Kordon and Janos Sztipanovits (ed.): "Networked Systems: Realization of Reliable Systems on Unreliable Networked Platforms," *Proceedings of the Sixth Monterey Workshop*, LNCS Vol. 4322, 2007
Networked computing is increasingly becoming the universal integrator for large-scale systems. In addition, new generation of wireless networked embedded systems rapidly create new technological environments that imply complex interdependencies amongst all layers of societal-scale critical infrastructure, such as transportation, energy distribution and telecommunication. This trend makes reliability and safety of networked computing a crucial issue and a technical precondition for building software intensive systems that are robust, fault tolerant, and highly available. The 12th Monterey Workshop on "Networked Systems: Realization of Reliable Systems on Unreliable Networked Platforms" focused on new, promising directions in achieving high software and system reliability in networked systems.

[24] Sztipanovits, J.: "Model-based Software Development," ESMD-SW Workshop, NASA, Houston, TX March 15-17, 2007
Model based software and system design is based on the end-to-end use of formal, composable and manipulable models in the product life-cycle. An emerging common thread is that modeling languages are domain-specific: they offer software developers concepts and notations that are tailored to capture essential characteristics of their application domain. Model Integrated Computing (MIC) developed at the Institute for Software Integrated Systems (ISIS) at Vanderbilt University is part of this new direction. The presentation provides an overview of key principles, methods and tools of model-based software and systems design and discusses application directions and experience in high-confidence systems design, architecture exploration and model-based systems integration.

[25] Sztipanovits, J.: "Defining Behavioral Semantics for Domain Specific Modeling Languages," NSF Workshop on Semantics for Event-Based Modeling, RTAS, Bellevue, WA, April 3, 2007.
Domain-Specific Modeling Languages (DSMLs) play fundamental role in the model-based design of embedded software and systems. While abstract syntax metamodeling enables the rapid and inexpensive development of DSMLs, the specification of DSML semantics is still a hard problem. In previous work, we have developed methods and tools for the semantic anchoring of DSML-s. Semantic anchoring introduces a set of reusable "semantic units" that provides reference semantics for basic behavioral categories using the Abstract State Machine framework. In this presentation, we extend the semantic anchoring framework to heterogeneous behaviors by developing method for the composition of semantic units. Semantic unit composition reduces the required effort from DSML designers and improves the quality of the specification. The proposed method is demonstrated through a case study.

[26] I. Roychoudhury, M. Daigle, G. Biswas, X. Koutsoukos, and P. J. Mosterman, "A Method for Efficient Simulation of Hybrid Bond Graphs," *International Conference*

on Bond Graph Modeling and Simulation (ICBGM 2007), pp. 177-184, Jan 2007.
The hybrid bond graph (HBG) paradigm is a uniform, multi-domain physics-based modeling language. It incorporates controlled and autonomous mode changes as idealized switching functions that enable the reconfiguration of energy flow paths to model hybrid physical systems. Building accurate and computationally efficient simulation mechanisms from HBG models is a challenging task, especially when there is no a priori knowledge of the subset of system modes that will be active during the simulation. In this work, we present an approach that exploits the inherent causal structure in HBG models to derive efficient hybrid simulation models as reconfigurable block diagram structures. We present a MATLAB/Simulink implementation of our approach and demonstrate its effectiveness using an electrical circuit example.

[27] M. Daigle, I. Roychoudhury, G. Biswas, and X. Koutsoukos, "Efficient Simulation of Component-Based Hybrid Models Represented as Hybrid Bond Graphs," *Hybrid Systems: Computation and Control (HSCC 2007)*, Lecture Notes in Computer Science, vol. 4416, pp. 680-683, Apr 2007.
The complexity of modern embedded systems often requires the use of simulations for systematic design, analysis, and verification tasks. The nonlinear and hybrid nature of these systems make the building of accurate and computationally efficient simulation models very challenging. In this work, we adopt the Hybrid Bond Graph (HBG) paradigm, a uniform, multi-domain physics-based modeling language with local switching functions that enable the reconfiguration of energy flow paths to model hybrid systems. The inherent causal structure in HBG models is exploited to derive efficient hybrid simulation models as reconfigurable block diagram structures. We demonstrate our approach by modeling and analyzing the behavior of an electrical power system.

[28] Antoon Goderisa, Christopher Brooks, lkay Altintas, Edward A. Lee, "Composing Different Models of Computation in Ptolemy II and Kepler," 2007 Proceedings, International Conference on Computational Science (ICCS), May, 2007; To appear at International Conference on Computational Science (ICCS) 2007.

A model of computation (MoC) is a formal abstraction of execution in a computer. There is a need for composing MoCs in e-science. Kepler, which is based on Ptolemy II, is a scientific workflow environment that allows for MoC composition. This paper explains how MoCs are combined in Kepler and Ptolemy II and analyzes which combinations of MoCs are currently possible and useful. It demonstrates the approach by combining MoCs involving dataflow and finite state machines. The resulting classification should be relevant to other workflow environments wishing to combine multiple MoCs. Keywords: Model of computation, scientific workflow, Kepler, Ptolemy II.

[29] Takashi Nagata, Masayoshi Tomizuka, "Engine Torque Control Based on Discrete Event Model and Disturbance Observer," ASME International Mechanical Engineering Congress and Exposition (IMECE2007), (submitted), May, 2007; Draft submitted in May 2007, to be presented in Nov. 2007.

This paper presents a novel control method for torque generation in spark ignition (SI) engine. A model-based approach is employed to control engine torque output by adjusting throttle air intake with considerations of robust stability and performance. Discrete event engine model (DEM) is adopted with an addition of torque generation dynamics. Disturbance observer (DOB) techniques are utilized to achieve robust stability and performance by regarding the discrepancy between the actual plant and the nominal plant with desired plant characteristics as an equivalent disturbance input, which is estimated and cancelled. The desired plant behavior is stably realized by the DOB up to a bandwidth sufficient for a torque control application covered in our previous works. Numerical results show the effectiveness of the proposed scheme.

[30] Gang Zhou, Man-Kit Leung, and Edward A. Lee, "A Code Generation Framework for Actor-Oriented Models with Partial Evaluation," Proceedings of International Conference on Embedded Software and Systems 2007, LNCS 4523, Y.-H. Lee et al., 786-799, May, 2007.

Embedded software requires concurrency formalisms other than threads and mutexes used in traditional programming languages like C. Actor-oriented design presents a high level abstraction for composing concurrent components. However, high level abstraction often introduces overhead and results in slower system. We address the problem of generating efficient implementation for the systems with such a high level description. We use partial evaluation as an optimized compilation technique for actor-oriented models. We use a helper-based mechanism, which results in flexible and extensible code generation framework. The end result is that the benefit offered by high level abstraction comes with (almost) no performance penalty. The code generation framework has been released in open source form as part of Ptolemy II 6.0.1.

[31] Jongho Lee, Mikael Eklund, Shankar Sastry, Unpublished article, "Group pursuit policies for UUVs using nonlinear model predictive control and reachable sets," May, 2007.

A decentralized control scheme for large packs of unmanned underwater vehicles (UUV) using reachability computation and model predictive control is proposed and investigated. This scheme fits within a broader template based method for organizing and controlling large groups of UUVs and addressed one important mode of operation for these vehicles, specifically decentralized, coordinated group pursuit and tracking of intruders through the UUV network. The reachability computation and model predictive control schemes are presented along with simulation results with medium-sized packs of UUVs. Hardware implementation is also described for five UUVs

[32] Arkadeb Ghosal, Sri Kanajan, Randall Urbance, Alberto Sangiovanni-Vincentelli, "An Initial Study on Monetary Cost Evaluation for the Design of Automotive Electrical Architectures," SAE, April, 2007.

One of the many challenges facing electronic system architects is how to provide a cost estimate related to design decisions over the entire life-cycle and product line of the

architecture. Various cost modeling techniques may be used to perform this estimation. However, the estimation is often done in an ad-hoc manner, based on specific design scenarios or business assumptions. This situation may yield an unfair comparison of architectural alternatives due to the limited scope of the evaluation. A preferred estimation method would involve rigorous cost modeling based on architectural design cost drivers similar to those used in the manufacturing (e.g. process-based technical cost modeling) or in the enterprise software domain (e.g. COCOMO). This paper describes an initial study of a cost model associated with automotive electronic system architecture. The model's intended use is to evaluate system cost drivers in response to various architectural decisions (e.g. choosing a communication bus topology or mapping a function to hardware). The primary cost driver categories explored are design and development, part fabrication, assembly and in-service costs. The preliminary version of this cost model focuses on describing the key influences on cost, but not the entire mathematical model. The paper presents the cost model with the help of influence diagrams and illustrates the use of the cost modeling methodology through an automotive case study – a steer-by-wire system. As future work, we propose to build a cost model and supporting methodology that accounts for architecture evolution to address the issue of evolving architecture requirements as well as when and where to employ new technology in the architecture.

[33] Nadathur Satish, Kaushik Ravindran and Kurt Keutzer, "A Decomposition-based Constraint Optimization Approach for Statically Scheduling Task Graphs with Communication Delays to Multiprocessors," 10th Conference of Design, Automation and Test in Europe (DATE-07), 230-235, April, 2007.

We present a decomposition strategy to speed up constraint optimization for a representative multiprocessor scheduling problem. In the manner of Benders decomposition, our technique solves relaxed versions of the problem and iteratively learns constraints to prune the solution space. Typical formulations suffer prohibitive run times even on medium-sized problems with less than 30 tasks. Our decomposition strategy enhances constraint optimization to robustly handle instances with over 100 tasks. Moreover, the extensibility of constraint formulations permits realistic application and resource constraints, which is a limitation of common heuristic methods for scheduling. The inherent extensibility, coupled with improved run times from a decomposition strategy, posit constraint optimization as a powerful tool for resource constrained scheduling and multiprocessor design space exploration.

[34] Thomas Huining Feng, Lynn Wang, Wei Zheng, Sri Kanajan, and Sanjit A. Seshia, "Automatic Model Generation for Black Box Real-Time Systems," Design, Automation and Test in Europe (DATE) Conference, April, 2007.

Embedded systems are often assembled from black box components. System-level analyses, including verification and timing analysis, typically assume the system description, such as RTL or source code, as an input. There is therefore a need to automatically generate formal models of black box components to facilitate analysis. We propose a new method to generate models of real-time embedded systems based on

machine learning from execution traces, under a given hypothesis about the system's model of computation. Our technique is based on a novel formulation of the model generation problem as learning a dependency graph that indicates partial ordering between tasks. Tests based on an industry case study demonstrate that the learning algorithm can scale up and that the deduced system model accurately reflects dependencies between tasks in the original design. These dependencies help us formally prove properties of the system and also extract data dependencies that are not explicitly stated in the specifications of black box components.

[35] Sumitra Ganesh, Aaron Ames, Ruzena Bajcsy, "Composition of Dynamical Systems for Estimation of Human Body Dynamics," Proceedings of 10th International Conference on Hybrid Systems Computation and Control 2007, Alberto Bemporad, Antonio Bicchi, Giorgio Buttazzo, 702-706, April, 2007.

(No abstract.)

[36] Wei Zheng, Marco Di Natale, Claudio Pinello, Paolo Giusto, Alberto Sangiovanni Vincentelli, "Synthesis of task and message activation models in real-time distributed automotive systems," Design, Automation and Test in Europe, April, 2007.

Modern automotive architectures support the execution of distributed safety- and time-critical functions on a complex networked system with several buses and tens of ECUs. Schedulability theory allows the analysis of the worst case end-to-end latencies and the evaluation of the possible architecture configurations options with respect to timing constraints. We present an optimization framework, based on an ILP formulation of the problem, to select the communication and synchronization model that exploits the trade-offs between the purely periodic and the precedence constrained data-driven activation models to meet the latency and jitter requirements of the application. We demonstrated its effectiveness by optimizing complex real-life GM architecture.

[37] Marco Di Natale, Wei Zheng, Claudio Pinello, Paolo Giusto, Alberto Sangiovanni-Vincentelli, "Optimizing end-to-end latencies by adaptation of the activation events in distributed automotive systems," Real-Time and Embedded Technology and Applications Symposium, April, 2007.

Schedulability theory provides support for the analysis of the worst case latencies in distributed computations when the architecture of the system is known and the communication and synchronization mechanisms have been defined. In the design of complex automotive systems, however, a great benefit of schedulability analysis may come from its use as an aid in the exploration of the software architecture configurations that can best support the target application. We present an optimization algorithm that leverages the trade-offs between the purely periodic and the data-driven activation models to meet the latency requirements of distributed vehicle functions. We demonstrate its effectiveness on a complex automotive architecture.

[38] Krishnendu Chatterjee and Thomas A. Henzinger, "Assume-guarantee Synthesis," TACAS, March, 2007.

The classical synthesis problem for reactive systems asks, given a proponent process A and an opponent process B, to refine A so that the closed-loop system A||B satisfies a given specification Φ. The solution of this problem requires the computation of a winning strategy for proponent A in a game against opponent B. We define and study the *co-synthesis* problem, where the proponent A consists itself of two independent processes, $A=A_1||A_2$, with specifications Φ_1 and Φ_2, and the goal is to refine both A and A_2 so that $A_1||A_2||B$ satisfies $\Phi_1 \wedge \Phi_2$. For example, if the opponent B is a fair scheduler for the two processes A_1 and A_2, and Φ_i specifies the requirements of mutual exclusion for A_i (e.g., starvation freedom), then the co-synthesis problem asks for the automatic synthesis of a mutual-exclusion protocol.

We show that co-synthesis defined classically, with the processes A_1 and A_2 either collaborating or competing, does not capture desirable solutions. Instead, the proper formulation of co-synthesis is the one where process A_1 competes with A_2 but not at the price of violating Φ_1, and vice versa. We call this *assume-guarantee synthesis* and show that it can be solved by computing secure-equilibrium strategies. In particular, from mutual-exclusion requirements the assume-guarantee synthesis algorithm automatically computes Peterson's protocol.

[39] Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, "Generalized Parity Games," FOSSACS 07, March, 2007.

We consider games where the winning conditions are disjunctions (or dually, conjunctions) of parity conditions; we call them generalized parity games. These winning conditions, while omega-regular, arise naturally when considering fair simulation between parity automata, secure equilibria for parity conditions, and determinization of Rabin automata. We show that these games retain the computational complexity of Rabin and Streett conditions; i.e., they are NP-complete and co-NP-complete, respectively. The (co-)NP-hardness is proved for the special case of a conjunction/disjunction of two parity conditions, which is the case that arises in fair simulation and secure equilibria. However, considering these games as Rabin or Streett games is not optimal. We give an exposition of Zielonka's algorithm when specialized to this kind of games. The complexity of solving these games for k parity objectives with d parities, n states, and m edges is $O(n^{2kd} * m) * (k* d)! /(d!^k)$, as compared to $O(n^{2kd} * m) * (k* d)!$ when these games are solved as Rabin/Streett games. We also extend the subexponential algorithm for solving parity games recently introduced by Jurdzinski, Patterson, and Zwick to generalized parity games. The resulting complexity of solving generalized parity games is $n^{O(\sqrt{n})} * (k* d)!/(d!^k)$. As a corollary we obtain an improved algorithm for Rabin and Streett games with d pairs, with time complexity $n^{O(\sqrt{n})} * d!$.

[40] Krishnendu Chatterjee, "Optimal Strategy Synthesis for Stochastic Muller Games," FOSSACS 07, March, 2007.

The theory of graph games with omega-regular winning conditions is the foundation for modeling and synthesizing reactive processes. In the case of stochastic reactive processes, the corresponding stochastic graph games have three players, two of them (System and Environment) behaving adversarially, and the third (Uncertainty) behaving probabilistically. We consider two problems for stochastic graph games: the qualitative problem asks for the set of states from which a player can win with probability 1 (almost-sure winning); and the quantitative problem asks for the maximal probability of winning (optimal winning) from each state. We consider omega-regular winning conditions formalized as Muller winning conditions. We present optimal memory bounds for pure almost-sure winning and optimal winning strategies in stochastic graph games with Muller winning conditions. We also present improved memory bounds for randomized almost-sure winning and optimal strategies.

[41] Alberto L. Sangiovanni-Vincentelli, *Proceedings of the IEEE*, 95(3):467-506, March 2007.

System-level design (SLD) is considered by many as the next frontier in electronic design automation (EDA). SLD means many things to different people since there is no wide agreement on a definition of the term. Academia, designers, and EDA experts have taken different avenues to attack the problem, for the most part springing from the basis of traditional EDA and trying to raise the level of abstraction at which integrated circuit designs are captured, analyzed, and synthesized from. However, my opinion is that this is just the tip of the iceberg of a much bigger problem that is common to all system industry. In particular, I believe that notwithstanding the obvious differences in the vertical industrial segments (for example, consumer, automotive, computing, and communication); there is a common underlying basis that can be explored. This basis may yield a novel EDA industry and even a novel engineering field that could bring substantial productivity gains not only to the semiconductor industry but to all system industries including industrial and automotive, communication and computing, avionics and building automation, space and agriculture, and health and security, in short, a real technical renaissance.

[42] Abhijit Davare, Douglas Densmore, Trevor Meyerowitz, Alessandro Pinto, Alberto Sangiovanni-Vincentelli, Guang Yang, Haibo Zeng, Qi Zhu, "A Next-Generation Design Framework for Platform-based Design," DVCon 2007, February, 2007.

The platform-based design methodology is based on the usage of formal modeling techniques, clearly defined abstraction levels and the separation of concerns to enable an effective design process. The METROPOLIS framework embodies the platform-based design methodology and has been applied to a number of case studies across multiple domains. Based on these experiences, we have identified three key features that need to be enhanced: heterogeneous IP import, orthogonalization of performance from behavior, and design space exploration. The next generation METRO II framework incorporates these advanced features. The main concepts underlying METRO II are described in this paper and illustrated with a small example.

[43] Joseph Gerard Makin and Alessandro Abate, Technical report, "A Neural Hybrid-System Model of the Basal Ganglia," EECS Department - University of California, at Berkeley, January, 2007.

The basal ganglia (BG) are a set of functionally related and structurally interconnected nuclei in the human brain which form part of a closed loop between cortex and thalamus, receiving input from the former and outputting to the latter. The BG have been implicated in motor control and cognitive switching tasks; in particular, it is believed that the BG function as a controller for motor tasks by selectively disinhibiting appropriate portions of the thalamus and hence activating, via a feedback loop, cortical regions. These switching behaviors are perforce discrete, whereas the underlying dynamics of neuron voltages and neurotransmitter levels are continuous-time, continuous state phenomena. To this end, we propose and simulate a hybrid automaton for modeling individual neurons that affords explicit representation of voltage discharges and discrete outputs along with continuous voltage dynamics within a single, elegant model; and which is amenable both to the construction of large networks—in particular the cortico-basalthalamic loops—and to analysis on such networks.

[44] An Initial Study on Monetary Cost Evaluation for the Design of Automotive Electrical Architectures, *SAE 2007 Transactions: Journal of Passenger Cars: Electronic and Electrical Systems*, January 2007.

One of the many challenges facing electronic system architects is how to provide a cost estimate related to design decisions over the entire life-cycle and product line of the architecture. Various cost modeling techniques may be used to perform this estimation. However, the estimation is often done in an ad-hoc manner, based on specific design scenarios or business assumptions. This situation may yield an unfair comparison of architectural alternatives due to the limited scope of the evaluation. A preferred estimation method would involve rigorous cost modeling based on architectural design cost drivers similar to those used in the manufacturing (e.g. process-based technical cost modeling) or in the enterprise software domain (e.g. COCOMO). This paper describes an initial study of a cost model associated with automotive electronic system architecture. The model's intended use is to evaluate system cost drivers in response to various architectural decisions (e.g. choosing a communication bus topology or mapping a function to hardware). The primary cost driver categories explored are design and development, part fabrication, assembly and in-service costs. The preliminary version of this cost model focuses on describing the key influences on cost, but not the entire mathematical model. The paper presents the cost model with the help of influence diagrams and illustrates the use of the cost modeling methodology through an automotive case study – a steer-by-wire system. As future work, we propose to build a cost model and supporting methodology that accounts for architecture evolution to address the issue of evolving architecture requirements as well as when and where to employ new technology in the architecture.

[45] Krishnendu Chatterjee and Thomas A. Henzinger, Unpublished article, "Value Iteration," January, 2007; A Survey Paper submitted for publication in "25 Years in Model Checking".

We survey value iteration algorithms on graphs. Such algorithms can be used for determining the existence of certain paths (model checking), the existence of certain strategies (game solving), and the probabilities of certain events (performance analysis). We classify the algorithms according to the value domain (Boolean, probabilistic, or quantitative); according to the graph structure (nondeterministic, probabilistic, or multi-player); according to the desired property of paths (Borel level 1, 2, or 3); and according to the alternation depth and convergence rate of fixpoint computations.

[46] Eleftherios Matsikoudis and Edward A. Lee, Unpublished article, "Feedback in Strictly Causal Systems," January, 2007.

We ask whether strictly causal components form well defined systems when arranged in feedback configurations. The standard interpretation for such configurations induces a fixed-point constraint on the function modeling the component involved. We bring together ideas from the areas of set theory, order theory, and theory of generalized ultrametric spaces to establish the existence of a unique fixed point for any such function constructively, what has resisted more traditional approaches and has been a much coveted albeit elusive goal.

[47] Thomas A. Henzinger, "Games, time, and probability: Graph models for system design and analysis," Proceedings of the 33rd International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM), Lecture Notes in Computer Science, Springer, January, 2007.

Digital technology is increasingly deployed in safety-critical situations. This calls for systematic design and verification methodologies that can cope with three major sources of system complexity: concurrency, real time, and uncertainty. We advocate a two-step process: formal modeling followed by algorithmic analysis (or, "model building" followed by "model checking"). We model the concurrent components of a reactive system as potential collaborators or adversaries in a multi-player game with temporal objectives, such as system safety. The real-time aspect of embedded systems requires models that combine discrete state transitions and continuous state evolutions. Uncertainty in the environment is naturally modeled by probabilistic state changes. As a result, we obtain three orthogonal extensions of the basic state-transition graph model for reactive systems --game graphs, timed graphs, and stochastic graphs-- as well as combinations thereof. In this short text, we provide a uniform exposition of the underlying definitions. For verification algorithms, we refer the reader to the literature.

[48] Alex A. Kurzhanskiy, Pravin Varaiya, "Ellipsoidal Techniques for Reachability Analysis of Discrete-Time Linear Systems," *IEEE Transactions Automatic Control*, 52(1):26-38, January 2007.

This paper describes the computation of reach sets for discrete-time linear control systems with time-varying coefficients and ellipsoidal bounds on the controls and initial conditions. The algorithms construct external and internal ellipsoidal approximations that touch the reach set boundary from outside and from inside. Recurrence relations describe the time evolution of these approximations. An essential part of the paper deals with singular discrete-time linear systems.

[49] Jeff Gray, Juha-Pekka Tolvanen, Steven Kelly, Aniruddha Gokhale, Sandeep Neema, and Jonathan Sprinkle, Paul A. Fishwick, "Domain-Specific Modeling (in CRC Handbook of Dynamic System Modeling)," 7, (in publication), CRC Press, 2007.

Since the inception of the software industry, modeling tools have been a core product offered by commercial vendors. In this chapter, the essential characteristics of DSM are presented, including a discussion regarding those domains that are most likely to benefit from DSM adoption. The chapter also contains a case study section where two different examples are presented in two different metamodeling tools. An overview of the history of metamodeling tools is also provided, as well as concluding comments.

[50] A. Abate S. Amin and M. Prandini and J. Lygeros and S. Sastry, A. Bemporad A. Bicchi and G. Buttazzo, "Computational Approaches to Reachability Analysis of Stochastic Hybrid Systems," 4-17, 4416, Springer Verlag, 2007.

This work investigates some of the computational issues involved in the solution of probabilistic reachability problems for discrete time, controlled stochastic hybrid systems. It is first argued that, under rather weak continuity assumptions on the stochastic kernels that characterize the dynamics of the system, the numerical solution of a discretized version of the probabilistic reachability problem is guaranteed to converge to the optimal one, as the discretization level decreases. With reference to a benchmark problem, it is then discussed how some of the structural properties of the hybrid system under study can be exploited to solve the probabilistic reachability problem more efficiently. Possible techniques that can increase the scale-up potential of the proposed numerical approximation scheme are suggested.

[51] A. Abate, A. D'Innocenzo, G. Pola, M.D. Di Benedetto and S. Sastry, A. Bemporad and A. Bicchi and G. Buttazzo, "The Concept of Deadlock and Livelock in Hybrid Control Systems," 628-632, 4416, Springer Verlag, 2007.

This short paper qualitatively introduces the definition of the concepts of Deadlock and Livelock for a general class of Hybrid Control Systems (HCS). Such a characterization hinges on three important aspects: firstly, the concept of composition of HCS; secondly, the general concept of specifications and their composition for HCS; finally, the dynamical structure and behaviors of HCS. The first aspect is introduced in a novel manner, including ideas from the literature of discrete transition systems and accounting for concepts such as that of dynamical feedback interconnection. The second point includes general properties that are of interest from a systems and control theory perspective. The third part categorizes the diverse and possibly pathological behaviors

that are distinctive of HCS. A first look at the problem of Deadlock and Livelock Verification concludes the manuscript.

[52] S. Amin and A. Abate and M. Prandini and J. Lygeros and S. Sastry, J. Hespanha and A. Tiwari, "Reachability analysis for controlled discrete time stochastic hybrid systems," 49-63, 3927, Springer Verlag, 2007.

A model for discrete time stochastic hybrid systems whose evolution can be influenced by some control input is proposed in this paper. With reference to the introduced class of systems, a methodology for probabilistic reachability analysis is developed that is relevant to safety verification. This methodology is based on the interpretation of the safety verification problem as an optimal control problem for a certain controlled Markov process. In particular, this allows characterizing through some optimal cost function the set of initial conditions for the system such that safety is guaranteed with sufficiently high probability. The proposed methodology is applied to the problem of regulating the average temperature in a room by a thermostat controlling a heater.

[53] Aaron D. Ames, "Homotopy Meaningful Hybrid Model Structures," American Mathematical Society, 2007.

Hybrid systems are systems that display both discrete and continuous behavior and, therefore, have the ability to model a wide range of robotic systems such as those undergoing impacts. The main observation of this paper is that systems of this form relate in a natural manner to very special diagrams over a category, termed hybrid objects. Using the theory of model categories, which provides a method for "doing homotopy theory" on general categories satisfying certain axioms, we are able to understand the homotopy theoretic properties of such hybrid objects in terms of their "non-hybrid" counterparts. Specifically, given a model category, we obtain a "homotopy meaningful" model structure on the category of hybrid objects over this category with the same discrete structure, i.e., a model structure that relates to the original non-hybrid model structure by means of homotopy colimits, which necessarily exist. This paper, therefore, lays the groundwork for "hybrid homotopy theory."

[54] A. D. Ames and S. Sastry, "Hybrid Geometric Reduction of Hybrid Systems," Submitted to the IEEE Conference on Decision and Control, December, 2006.

This paper presents a unifying framework in which to carry out the hybrid geometric reduction of hybrid systems, generalizing classical reduction to a hybrid setting. Utilizing hybrid category theory, all of the major ingredients necessary for classical reduction can be hybridized through the notion of a hybrid object and a hybrid morphism over a general category. By leveraging the results of Marsden and Weinstein, we are able to show that when there is a hybrid symplectic manifold (the hybrid phase space) on which a hybrid Lie group acts symplectically, we can reduce the hybrid phase space to another hybrid symplectic manifold in which the hybrid symmetries are "divided out." In addition, hybrid trajectories of a hybrid Hamiltonian on the hybrid phase space determine corresponding hybrid trajectories on the reduced hybrid space.

[55] Alessandro Abate, Ashish Tiwari and S. Shankar Sastry, Technical report, "The concept of Box Invariance for biologically-inspired dynamical systems," EECS Department - University of California, at Berkeley, December, 2006.

In this paper we introduce a special notion of Invariance Set for certain classes of dynamical systems: the concept has been inspired by our experience with models drawn from Biology. We claim that Box Invariance, that is, the existence of "boxed" invariant regions, is a characteristic of many biologically-inspired dynamical models, especially those derived from stoichiometric reactions. Moreover, box invariance is quite useful for the verification of safety properties of such systems. This paper presents effective characterization of this notion for linear and affine systems, the study of the dynamical properties it subsumes, computational aspects of checking for box invariance, and a comparison with related concepts in the literature. The concept is illustrated using two models from biology.

[56] Allen Y. Yang, John Wright, Shankar Sastry, and Yi Ma, Technical report, "Unsupervised Segmentation of Natural Images via Lossy Data Compression," UC Berkeley, UCB/EECS-2006-195, December, 2006.

In this paper, we cast natural-image segmentation as a problem of clustering texture features as multivariate mixed data. We model the distribution of the texture features using a mixture of Gaussian distributions. However, unlike most existing clustering methods, we allow the mixture components to be degenerate or nearly-degenerate. We contend that this assumption is particularly important for mid-level image segmentation, where degeneracy is typically introduced by using a common feature representation for different textures. We show that such a mixture distribution can be effectively segmented by a simple agglomerative clustering algorithm derived from a lossy data compression approach. Using simple fixed-size Gaussian windows as texture features, the algorithm segments an image by minimizing the overall coding length of all the feature vectors. In terms of a variety of performance indices, our algorithm compares favorably against other well-known image segmentation methods on the Berkeley image database.

[57] Ye Zhou and Edward A. Lee, "A Causality Interface for Deadlock Analysis in Dataflow," EMSOFT 2006, October, 2006.

In this paper, we consider a concurrent model of computation called dataflow, where components (actors) communicate via streams of data tokens. Dataflow semantics has been adopted by experimental and production languages used to design embedded systems. The execution of a data-flow actor is enabled by the availability of its input data. One important question is whether a dataflow model will deadlock (i.e., actors cannot execute due to a data dependency loop). Deadlock in many cases can be determined, although it is generally not decidable. We develop a causality interface for dataflow actors based on the general framework we introduced in [1] and show how this causality information can be algebraically composed so that composition of components acquire causality interfaces that are inferred from their components and the interconnections. We illustrate the use of these causality interfaces to statically analyze for deadlock.

[58] Abhijit Davare, Jike Chong, Qi Zhu, Douglas Densmore and Alberto Sangiovanni-Vincentelli, "An Overlap-based MILP Formulation for Task Allocation and Scheduling," submitted, October, 2006.

The deployment of applications on heterogeneous multi-core platforms is one of the most important challenges in the embedded systems design flow. In this paper, we develop an automated approach to tackle this problem that uses an efficient and extensible Mixed Integer Linear Programming (MILP) formulation. We show the effectiveness of our approach with extensive computational testing. For a case study involving the deployment of a Motion JPEG encoder application onto a Xilinx Virtex II Pro FPGA platform, we demonstrate that our automated approach can accurately capture the design space and yield systems that are competitive with manual designs.

[59] Arkadeb Ghosal, Thomas A. Henzinger, Daniel Iercan, Christoph Kirsch, Alberto Sangiovanni-Vincentelli., " A Hierarchical Coordination Language for Interacting Real-Time Tasks," EMSOFT 2006, October, 2006.

We designed and implemented a new programming language called Hierarchical Timing Language (HTL) for hard real-time systems. Critical timing constraints are specified within the language, and ensured by the compiler. Programs in HTL are extensible in two dimensions without changing their timing behavior: new program modules can be added, and individual program tasks can be refined. The mechanism supporting time invariance under parallel composition is that different program modules communicate at specified instances of time. Time invariance under refinement is achieved by conservative scheduling of the top level. HTL is a coordination language, in that individual tasks can be implemented in "foreign" languages. As a case study, we present a distributed HTL implementation of an automotive steer-by-wire controller.

[60] , "6th OOPSLA Workshop On Domain-Specific Modeling," Juha-Pekka Tolvanen, Matti Rossi, Jonathan Sprinkle, University of Jyvaskala, October, 2006.

The 6th DSM workshop featured 22 research and position papers describing new ideas at either a practical or theoretical level. On the practical side, several papers described application of modeling techniques within a specific domain. In addition to industrial projects, several authors from academia presented research ideas that initiate and forward the technical underpinnings of domain-specific modeling. Workshop had over 40 participants.

[61] Bor-Yuh Evan Chang, Matthew Harren, and George C. Necula, "Analysis of Low-Level Code Using Cooperating Decompilers," The 13th International Static Analysis Symposium (SAS), Kwangkeun Yi, 318-335, September, 2006;

Analysis or verification of low-level code is useful for minimizing the disconnect between what is verified and what is actually executed and is necessary when source code is unavailable or is, say, intermingled with inline assembly. We present a modular framework for building pipelines of cooperating decompilers that gradually lift the level

of the language to something appropriate for source-level tools. Each decompilation stage contains an abstract interpreter that encapsulates its findings about the program by translating the program into a higher-level intermediate language. We provide evidence for the modularity of this framework through the implementation of multiple decompilation pipelines for both x86 and MIPS assembly produced by gcc, gcj, and coolc (a compiler for a pedagogical Java-like language) that share several low-level components. Finally, we discuss our experimental results that apply the BLAST model checker for C and the Cqual analyzer to decompiled assembly.

[62] Krishnendu Chatterjee, Luca de Alfaro, Marco Faella, Thomas A. Henzinger, Rupak Majumdar and Marielle Stoelinga, "Quantitative Compositional Pricing," QEST 06, September, 2006.

We present a compositional theory of system verification, where specifications assign real-numbered costs to systems. These costs can express a wide variety of quantitative system properties, such as resource consumption, price, or a measure of how well a system satisfies its specification. The theory supports the composition of systems and specifications, and the hiding of variables. Boolean refinement relations are replaced by real-numbered distances between descriptions of a system at different levels of detail. We show that the classical Boolean rules for compositional reasoning have quantitative counterparts in our setting. While our general theory allows costs to be specified by arbitrary cost functions, we also consider a class of linear cost functions, which give rise to an instance of our framework where all operations are computable in polynomial time.

[63] Krishnendu Chatterjee, Luca de Alfaro and Thomas A. Henzinger, "Strategy Improvement for Concurrent Reachability Games," QEST 06, September, 2006.

A concurrent reachability game is a two-player game played on a graph: at each state, the players simultaneously and independently select moves; the two moves determine jointly a probability distribution over the successor states. The objective for player 1 consists in reaching a set of target states; the objective for player 2 is to prevent this, so that the game is zero-sum. Our contributions are two-fold. First, we present a simple proof of the fact that in concurrent reachability games, for all epsilon>0, memoryless epsilon-optimal strategies exist. A memoryless strategy is independent of the history of plays, and an epsilon-optimal strategy achieves the objective with probability within epsilon of the value of the game. In contrast to previous proofs of this fact, which rely on the limit behavior of discounted games using advanced Puisieux series analysis, our proof is elementary and combinatorial. Second, we present a strategy-improvement (a.k.a. policy-iteration) algorithm for concurrent games with reachability objectives.

[64] Krishnendu Chatterjee, "Concurrent Games with Tail Objectives," CSL 06, September, 2006.

We study infinite stochastic games played by two-players over a finite state space, with objectives specified by sets of infinite traces. The games are concurrent (players make moves simultaneously and independently), stochastic (the next state is determined by a

probability distribution that depends on the current state and chosen moves of the players) and infinite (proceeds for infinite number of rounds). The analysis of concurrent stochastic games can be classified into: quantitative analysis, analyzing the optimum value of the game; and qualitative analysis, analyzing the set of states with optimum value 1. We consider concurrent games with tail objectives, i.e., objectives that are independent of the finite-prefix of traces, and show that the class of tail objectives is strictly richer than the omega-regular objectives. We develop new proof techniques to extend several properties of concurrent games with omega-regular objectives to concurrent games with tail objectives. We prove the positive limit-one property for tail objectives, that states for all concurrent games if the optimum value for a player is positive for a tail objective Phi at some state, then there is a state where the optimum value is 1 for Phi, for the player. We also show that the optimum values of zero-sum (strictly conflicting objectives) games with tail objectives can be related to equilibrium values of nonzero-sum (not strictly conflicting objectives) games with simpler reachability objectives. A consequence of our analysis presents a polynomial time reduction of the quantitative analysis of tail objectives to the qualitative analysis for the sub-class of one-player stochastic games (Markov decision processes).

[65] Krishnendu Chatterjee, "Nash Equilibrium for Upward-Closed Objectives," CSL 06, September, 2006.

We study infinite stochastic games played by n-players on a finite graph with goals specified by sets of infinite traces. The games are concurrent (each player simultaneously and independently chooses an action at each round), stochastic (the next state is determined by a probability distribution depending on the current state and the chosen actions), infinite (the game continues for an infinite number of rounds), nonzero-sum (the players' goals are not necessarily conflicting), and undiscounted. We show that if each player has an upward-closed objective, then there exists an epsilon-Nash equilibrium in memory less strategies, for every epsilon>0; and exact Nash equilibria need not exist. Upward-closure of an objective means that if a set Z of infinitely repeating states is winning, then all supersets of Z of infinitely repeating states are also winning. Memoryless strategies are strategies that are independent of history of plays and depend only on the current state. We also study the complexity of finding values (payoff profile) of an epsilon-Nash equilibrium. We show that the values of an epsilon-Nash equilibrium in nonzero-sum concurrent games with upward-closed objectives for all players can be computed by computing epsilon-Nash equilibrium values of nonzero-sum concurrent games with reachability objectives for all players and a polynomial procedure. As a consequence we establish that values of an epsilon-Nash equilibrium can be computed in TFNP (total functional NP), and hence in EXPTIME.

[66] Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger and Jean-Francois Raskin, "Algorithms for Omega-Regular Games with Imperfect Information," CSL 06, September, 2006.

We study observation-based strategies for two-player turn-based games on graphs with omega-regular objectives. An observation-based strategy relies on imperfect information

about the history of a play, namely, on the past sequence of observations. Such games occur in the synthesis of a controller that does not see the private state of the plant. Our main results are twofold. First, we give a fixed-point algorithm for computing the set of states from which a player can win with a deterministic observation-based strategy for any omega-regular objective. The fixed point is computed in the lattice of antichains of state sets. This algorithm has the advantages of being directed by the objective and of avoiding an explicit subset construction on the game graph. Second, we give an algorithm for computing the set of states from which a player can win with probability 1 with a randomized observation-based strategy for a Buchi objective. This set is of interest because in the absence of perfect information, randomized strategies are more powerful than deterministic ones. We show that our algorithms are optimal by proving matching lower bounds.

[67] Thomas A. Henzinger and Vinayak Prabhu, "Timed alternating-time temporal logic," FORMATS 2006, September, 2006.

We add freeze quantifiers to the game logic ATL in order to specify real-time objectives for games played on timed structures. We define the semantics of the resulting logic TATL by restricting the players to physically meaningful strategies, which do not prevent time from diverging. We show that TATL can be model checked over timed automaton games. We also specify timed optimization problems for physically meaningful strategies, and we show that for timed automaton games, the optimal answers can be approximated to within any degree of precision.

[68] Pannag R Sanketi, J. Carlos Zavala, J. K. Hedrick, M. Wilcutts, T. Kaga, "A Simplified Catalytic Converter Model for Automotive Coldstart Applications with Adaptive Parameter Fitting," 8th International Symposium on Advanced Vehicle Control, August, 2006; .

It is well known that a major portion of the unburned hydrocarbon (HC) emissions in a typical drive cycle of an automotive engine are produced in the initial 1-2 minutes of operation, commonly called as the "coldstart" period. Catalyst light-off is essential for reducing these emissions. Model-based paradigm is used to develop a control-oriented, thermodynamics based simple catalyst model for coldstart analysis. The catalyst thermal submodel is modeled as a hybrid system consisting of three discrete states and one continuous state. The discrete states are "initial warm-up", "evaporation of condensed gas" and "light-off". The continuous dynamics consists of the catalyst temperature. In each of the discrete states, energy balance of a control mass is used to model the catalyst temperature. Parameter adaptation algorithm (PAA) is used to identify the parameters in each of the discrete states. Effectiveness of the average values of the adjusted parameters is also given. Wiebe profiles are adopted to empirically model the HC emissions conversion properties of the catalyst as a function of the catalyst temperature and the air-fuel ratio. The static efficiency maps are further extended to include the effects of spatial velocity of the feedgas. Experimental results indicate good agreement with the model estimates for the catalyst warm-up.

[69] Takashi Nagata, Hwan Hur, Masayoshi Tomizuka, "Model-Based Control for Smooth Gear Shifting by Engine-AT Collaboration," 8th International Symposium on Advanced Vehicle Control (AVEC '06), 635-640, August, 2006.

A collaborative control scheme between engine and automatic transmission (AT) gearbox is proposed to realize a smoother gear shifting for improving riding comfort. A systematic procedure is developed to obtain a desired engine torque profile and a desired hydraulics actuation profile for the AT gearbox which will minimize shocks due to gear shifting. Tracking controls must be applied to both the engine and the AT gearbox to follow these profiles. For the model-based approach, an AT gearbox model is devised using hybrid systems techniques to represent the discrete-event nature of automatic transmissions. Numerical evaluations have shown the effectiveness of the proposed scheme.

[70] Mark L. McKelvin, Jr., Claudio Pinello, Sri Kanajan, Joseph Wysocki, Alberto Sangiovanni-Vincentelli, "Model-Based Design of Heterogeneous Systems for Fault Tree Analysis," 24th International System Safety Conference, Rodney J. Simmons, Ph.D., Norman J. Gauthier, System Safety Society, 400-409, August, 2006.

We introduce a model-based approach to heterogeneous system design that enables the automatic generation of fault trees for analyzing system reliability properties. This approach extends our previous work that addressed the generation of fault trees from a dataflow model. In this new context, heterogeneous systems are composed of interacting discrete-time components, such as an electronic feedback controller, and continuous-time components, such as a plant. More recent work in computer-aided fault-tree generation methods is based on functional models of the system to produce a system fault tree automatically. Yet, most of these approaches were not applied to heterogeneous systems. Furthermore, these approaches continued to rely on intuition to create fault trees. Since in this approach fault tree generation is disjoint from the system modeling, consistency problems may arise when the structure and behavior of the system model is not accurately reflected. Our approach is different since we use a model of the system specified as a set of mathematical equations to derive the system fault modes and ultimately produce fault trees for heterogeneous systems.

[71] Krishnendu Chatterjee and Thomas A. Henzinger, "Strategy Improvement for Stochastic Rabin and Streett Games," CONCUR 06, August, 2006.

A stochastic graph game is played by two players on a game graph with probabilistic transitions. We consider stochastic graph games with omega-regular winning conditions specified as Rabin or Streett objectives. These games are NP-complete and coNP-complete, respectively. The value of the game for a player at a state s given an objective Phi is the maximal probability with which the player can guarantee the satisfaction of Phi from s. We present a strategy-improvement algorithm to compute values in stochastic Rabin games, where an improvement step involves solving Markov decision processes (MDPs) and nonstochastic Rabin games. The algorithm also computes values for stochastic Streett games but does not directly yield an optimal strategy for Streett

objectives. We then show how to obtain an optimal strategy for Streett objectives by solving certain nonstochastic Streett games.

[72] Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, "Algorithms for Buchi Games," GDV 06, August, 2006.

The classical algorithm for solving Buchi games requires time O(n * m) for game graphs with n states and m edges. For game graphs with constant outdegree, the best known algorithm has running time O(n*n /log n). We present two new algorithms for Buchi games. First, we give an algorithm that performs at most O(m) more work than the classical algorithm, but runs in time O(n) on infinitely many graphs of constant outdegree on which the classical algorithm requires time O(n*n). Second, we give an algorithm with running time O(n * m *log delta(n)/log n), where 1<= delta(n) <= n, is the outdegree of the game graph. Note that this algorithm performs asymptotically better than the classical algorithm if delta(n)=O(log n).

[73] Xiaojun Liu, Eleftherios Matsikoudis, and Edward A. Lee, "Modeling Timed Concurrent Systems," CONCUR 2006 - Concurrency Theory, 17th International Conference, Christel Baier and Holger Hermanns, 1-15, August, 2006.

Timed concurrent systems are widely used in concurrent and distributed real-time software, modeling of hybrid systems, design of hardware systems (using hardware description languages), discrete-event simulation, and modeling of communication networks. They consist of concurrent components that communicate using timed signals, that is, sets of (semantically) time-stamped events. The denotational semantics of such systems is traditionally formulated in a metric space, wherein causal components are modeled as contracting functions. We show that this formulation excessively restricts the models of time that can be used. In particular, it cannot handle super-dense time, commonly used in hardware description languages and hybrid systems modeling, finite time lines, and time with no origin. Moreover, if we admit continuous time and mixed signals (essential for hybrid systems modeling) or certain Zeno signals, then causality is no longer equivalent to its formalization in terms of contracting functions. In this paper, we offer an alternative semantic framework using a generalized ultrametric that overcomes these limitations.

[74] Thomas A. Henzinger and Joseph Sifakis, "The embedded systems design challenge," Proceedings of the 14th International Symposium on Formal Methods (FM), Lecture Notes in Computer Science, Springer, August, 2006.

We summarize some current trends in embedded systems design and point out some of their characteristics, such as the chasm between analytical and computational models, and the gap between safety-critical and best-effort engineering practices. We call for a coherent scientific foundation for embedded systems design, and we discuss a few key demands on such a foundation: the need for encompassing several manifestations of heterogeneity, and the need for constructivity in design. We believe that the development

of a satisfactory Embedded Systems Design Science provides a timely challenge and opportunity for reinvigorating computer science.

[75] Qi Zhu, Abhijit Davare and Alberto Sangiovanni-Vincentelli, "A Semantic-Driven Synthesis Flow for Platform-Based Design," submitted to Fourth ACM-IEEE International Conference on Formal Methods and Models for Codesign (MEMOCODE'06), July, 2006.

The separation of concerns between functionality and architecture is a powerful technique used to facilitate design reuse at the system level. This separation of concerns and the successive refinement of the design by mapping functionality onto architecture are the core concepts in platform-based design. The goal of mapping is to optimize a set of objective functions while satisfying constraints on the mapped design. The mapping step can be seen as a synthesis process. While applying the methodology to a number of case studies, we have found that to gain the benefits of correct-by-construction deployment and rapid design space exploration, neither the semantics nor the abstraction levels for modeling can be chosen in an ad-hoc manner. In this paper, we propose a semantics-driven synthesis flow, in which the abstraction level and operational semantics are determined formally by using the concept of a common semantic domain between functionality and architecture. By doing so, a formal synthesis procedure can be defined and algorithms for automatic optimal mapping derived. By applying this approach to the previously mentioned case studies, we demonstrate how this approach can be used to significantly improve the effectiveness of platform-based design.

[76] A. D. Ames, R. D. Gregg, E. D. B. Wendel and S. Sastry, "Towards the Geometric Reduction of Controlled Three-Dimensional Robotic Bipedal Walkers," Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, July, 2006.

The purpose of this paper is to apply methods from geometric mechanics to the analysis and control of robotic bipedal walkers. We begin by introducing a generalization of Routhian reduction, functional Routhian Reduction, which allows for the conserved quantities to be functions of the cyclic variables rather than constants. Since bipedal robotic walkers are naturally modeled as hybrid systems, which are inherently nonsmooth, in order to apply this framework to these systems it is necessary to first extend functional Routhian reduction to a hybrid setting. We apply this extension, along with potential shaping and controlled symmetries, to derive a feedback control law for a three-dimensional bipedal walker that provably results in walking gaits on flat ground.

[77] A. Abate, M. Chen and S. Sastry, "Analysis of an Implementable Application Layer Scheme for Flow Control over Wireless Networks," Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, July, 2006.

This paper deals with the problem of congestion control and packet exchange on a wireless network. The mathematical model of the protocol is inspired by and extends a known fluid flow scheme for the control of congestion on a wired network. The necessity to introduce a specific wireless model is motivated by the presence of channel error; often

this error (due to intrinsic noise or channel corruption) is not known exactly. This motivates the approximation of parts of the structure of the model with binary functions, whose switching point can be precisely known. These new discontinuous elements, while in practice greatly simplifying the structure of the algorithm (they carry a single bit of information), complicate the theoretical analysis of its dynamical properties. We therefore approximate them with continuous functions with proper limiting behavior: they thus preserver the simple shape and yield themselves to analysis as well. Given this setup, we then investigate the important issues of existence and uniqueness of the equilibrium for the dynamical system, and of local asymptotic stability. Furthermore, we show that this equilibrium solves a concave net utility optimization problem, of which the classical one for wired networks is a special case. The take away point of this work is that the scheme we propose to handle the traffic on a wireless network is not only innovative and meaningful, but has also the potential to be modified and translated into practical implementations.

[78] Adam Cataldo, Edward Lee, Xiaojun Liu, Eleftherios Matsikoudis, and Haiyang Zheng, "A Constructive Fixed-Point Theorem and the Feedback Semantics of Timed Systems," Workshop on Discrete Event Systems, July, 2006.

Deterministic timed systems can be modeled as fixed point problems. In particular, any connected network of timed systems can be modeled as a single system with feedback, and the system behavior is the fixed point of the corresponding system equation, when it exists. For delta-causal systems, we can use the Cantor metric to measure the distance between signals and the Banach fixed-point theorem to prove the existence and uniqueness of a system behavior. Moreover, the Banach fixed-point theorem is constructive: it provides a method to construct the unique fixed point through iteration. In this paper, we extend this result to systems modeled with the superdense model of time used in hybrid systems. We call the systems we consider eventually delta-causal, a strict generalization of delta-causal in which multiple events may be generated on a signal in zero time. With this model of time, we can use a generalized ultrametric instead of a metric to model the distance between signals. The existence and uniqueness of behaviors for such systems comes from the fixed-point theorem of Priess-Crampe, but this theorem gives no constructive method to compute the fixed point. This leads us to define petrics, a generalization of metrics, which we use to generalize the Banach fixed-point theorem to provide a constructive fixed-point theorem. This new fixed-point theorem allows us to construct the unique behavior of eventually delta-causal systems

[79] Aaron D. Ames, Robert D. Gregg, Eric D.B. Wendel and Shankar Sastry, "Towards the Geometric Reduction of Controlled Three-Dimensional Bipedal Robotic Walkers," 3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, July, 2006.

The purpose of this paper is to apply methods from geometric mechanics to the analysis and control of bipedal robotic walkers. We begin by introducing a generalization of Routhian reduction, functional Routhian Reduction, which allows for the conserved quantities to be functions of the cyclic variables rather than constants. Since bipedal

robotic walkers are naturally modeled as hybrid systems, which are inherently nonsmooth, in order to apply this framework to these systems it is necessary to first extend functional Routhian reduction to a hybrid setting. We apply this extension, along with potential shaping and controlled symmetries, to derive a feedback control law that provably results in walking gaits on flat ground for a three-dimensional bipedal walker given walking gaits in two-dimensions.

[80] Aaron D. Ames, Robert D. Gregg, Haiyang Zheng, Prof. Shankar Sastry, "Is There Life After Zeno? Taking Executions past the Breaking (Zeno) Point," 2007 American Control Conference, ACS, June, 2006; Presented in conjunction with BEARS 2006.

In this paper we propose a technique to extend the simulation of a Zeno hybrid system beyond its Zeno time point. A Zeno hybrid system model is a hybrid system with an execution that takes an infinite number of discrete transitions during a finite time interval. We argue that the presence of Zeno behavior indicates that the hybrid system model is incomplete by considering some classical Zeno models that incompletely describe the dynamics of the system being modeled. This motivates the systematic development of a method for completing hybrid system models through the introduction of new post-Zeno states, where the completed hybrid system transitions to these post-Zeno states at the Zeno time point. In practice, simulating a Zeno hybrid system is challenging in that simulation effectively halts near the Zeno time point. Moreover, due to unavoidable numerical errors, it is not practical to exactly simulate a Zeno hybrid system. Therefore, we propose a method for constructing approximations of Zeno models by leveraging the completed hybrid system model. Using these approximations, we can simulate a Zeno hybrid system model beyond its Zeno point and reveal the complete dynamics of the system being modeled.

[81] Shinjiro Kakita, Yosinori Watanabe, Douglas Densmore, Abhijit Davare, Alberto Sangiovanni-Vincentelli, "Functional Model Exploration for Multimedia Applications via Algebraic Operators," ACSD 2006 - Sixth International Conference on Application of Concurrency to System Design, June, 2006.

An optimized functional design space exploration method for multimedia applications is proposed. The basis of the method is a way of representing the dependency and the concurrency of an application in a compact form exploiting algebraic operators and expressions. The optimized design process consists of mapping one of the possible expressions in the application space onto a concurrent architecture. We use the Metropolis design framework to demonstrate the effectiveness of the procedure using an FPGA architecture as the target implementation platform. The advantage of using this platform is the availability of models that approximate well the performance of the final implementation when performing the mapping from function to architecture thus yielding a robust design methodology.

[82] A. D. Ames and S. Sastry, " Hybrid Routhian Reduction of Lagrangian Hybrid Systems," American Control Conference, June, 2006.

This paper extends Routhian reduction to a hybrid setting, i.e., to systems that display both continuous and discrete behavior. We begin by considering a Lagrangian together with a configuration space with unilateral constraints on the set of admissible configurations. This naturally yields the notion of a hybrid Lagrangian, from which we obtain a Lagrangian hybrid system in a way analogous to the association of a Lagrangian vector field to a Lagrangian. We first give general conditions on when it is possible to reduce a cyclic Lagrangian hybrid system, and explicitly compute the reduced Lagrangian hybrid system in the case when it is obtained from a cyclic hybrid Lagrangian.

[83] A. D. Ames and S. Sastry, "Hybrid Cotangent Bundle Reduction of Simple Hybrid Mechanical Systems with Symmetry," American Control Conference, June, 2006.

This paper begins by introducing the notion of a simple hybrid mechanical system, which generalizes mechanical systems to include unilateral constraints on the configuration space. From such a system we obtain, explicitly, a simple hybrid system. The main contribution of this paper is to provide conditions on when it is possible to reduce the phase space of hybrid systems obtained from simple hybrid mechanical systems, and general simple hybrid systems, due to symmetries in the systems. Specifically, given a Hamiltonian G-space---which is the ingredient needed to reduce continuous systems---we find conditions on the hybrid system and the G-space so that reduction can be carried out in a hybrid setting---conditions that are explicitly related to conditions on the original hybrid mechanical system.

[84] A. D. Ames, H. Zheng, R. D. Gregg and S. Sastry, "Is there Life after Zeno? Taking Executions Past the Breaking (Zeno) Point," American Control Conference, June, 2006.

Understanding Zeno phenomena plays an important role in understanding hybrid systems. A natural---and intriguing---question to ask is: what happens after a Zeno point? Inspired by the construction of Filippov, we propose a method for extending Zeno executions past a Zeno point for a class of hybrid systems: Lagrangian hybrid systems. We argue that after the Zeno point is reached, the hybrid system should switch to a holonomically constrained dynamical system, where the holonomic constraints are based on the unilateral constraints on the configuration space that originally defined the hybrid system. These principles are substantiated with a series of examples.

[85] A. Abate, A. Ames and S. Sastry, "Error Bounds Based Stochastic Approximations and Simulations of Hybrid Dynamical Systems," Proceedings of the 25th American Control Conference, June, 2006.

This paper introduces, develops and discusses an integration-inspired methodology for the simulation and analysis of deterministic hybrid dynamical systems. When simulating hybrid systems, and thus unavoidably introducing some numerical error, a progressive tracking of this error can be exploited to discern the properties of the system, i.e., it can be used to introduce a stochastic approximation of the original hybrid system, the

simulation of which would give a more complete representation of the possible trajectories of the system. Moreover, the error can be controlled to check and even guarantee (in certain special cases) the robustness of simulated hybrid trajectories.

[86] A. Abate, A. Ames and S. Sastry, "A-Priori Detection of Zeno Behavior in Communication Networks Modeled as Hybrid Systems," Proceedings of the 25th American Control Conference, June, 2006.

In this paper, we show that the sufficient conditions for the existence of Zeno behavior in hybrid systems correctly predict such executions in a modeling instance of the fluid-flow approximation of the TCP-like protocol for wireless communication networks.

[87] A. Abate and A. Tiwari, "Box Invariance of hybrid and switched systems," Proceedings of the 2nd IFAC Conference on Analysis and Design of Hybrid Systems, IFAC, June, 2006.

This paper investigates the concept of box invariance for classes of hybrid and switched systems. After motivating and defining the notion, we present a concise summary of results on its characterization for single-domain dynamical systems. The notion is then extended to the case of hybrid and switched systems. We provide sufficient conditions for a hybrid or switched system to be box invariant. Models of many real systems, especially those drawn from biology, have been found to be box invariant. This paper illustrates the concept using a pharmacodynamic model of blood glucose metabolism.

[88] Aaron D. Ames and Shankar Sastry, "Hybrid Routhian Reduction of Lagrangian Hybrid Systems," American Control Conference, June, 2006.

This paper extends Routhian reduction to a hybrid setting, i.e., to systems that display both continuous and discrete behavior. We begin by considering a Lagrangian together with a configuration space with unilateral constraints on the set of admissible configurations. This naturally yields the notion of a hybrid Lagrangian, from which we obtain a Lagrangian hybrid system in a way analogous to the association of a Lagrangian vector field to a Lagrangian. We first give general conditions on when it is possible to reduce a cyclic Lagrangian hybrid system, and explicitly compute the reduced Lagrangian hybrid system in the case when it is obtained from a cyclic hybrid Lagrangian.

[89] Aaron D. Ames and Shankar Sastry, "Hybrid Cotangent Bundle Reduction of Simple Hybrid Mechanical Systems with Symmetry," American Control Conference, June, 2006.

This paper begins by introducing the notion of a simple hybrid mechanical system, which generalizes mechanical systems to include unilateral constraints on the configuration space. From such a system we obtain, explicitly, a simple hybrid system. The main contribution of this paper is to provide conditions on when it is possible to reduce the phase space of hybrid systems obtained from simple hybrid mechanical systems, and

general simple hybrid systems, due to symmetries in the systems. Specifically, given a Hamiltonian G-space—-which is the ingredient needed to reduce continuous systems—-we find conditions on the hybrid system and the G-space so that reduction can be carried out in a hybrid setting—-conditions that are explicitly related to conditions on the original hybrid mechanical system.

[90] Aaron D. Ames, Haiyang Zheng, Robert D. Gregg and Shankar Sastry, "Is there Life after Zeno? Taking Executions Past the Breaking (Zeno) Point," American Control Conference, June, 2006.

Understanding Zeno phenomena plays an important role in understanding hybrid systems. A natural and intriguing question to ask is: what happens after a Zeno point? Inspired by the construction of Filippov, we propose a method for extending Zeno executions past a Zeno point for a class of hybrid systems: Lagrangian hybrid systems. We argue that after the Zeno point is reached, the hybrid system should switch to a holonomically constrained dynamical system, where the holonomic constraints are based on the unilateral constraints on the configuration space that originally defined the hybrid system. These principles are substantiated with a series of examples.

[91] Haiyang Zheng, Edward A. Lee and Aaron D. Ames, Joao Hespanha, Ashish Tiwari, "Beyond Zeno: Get on with It!," Lecture Notes in Computer Science, Springer Berlin/Heidelberg, 3927, 2006, 3-540-33170-0.

In this paper we propose a technique to extend the simulation of a Zeno hybrid system beyond its Zeno time point. A Zeno hybrid system model is a hybrid system with an execution that takes an infinite number of discrete transitions during a finite time interval. We argue that the presence of Zeno behavior indicates that the hybrid system model is incomplete by considering some classical Zeno models that incompletely describe the dynamics of the system being modeled. This motivates the systematic development of a method for completing hybrid system models through the introduction of new post-Zeno states, where the completed hybrid system transitions to these post-Zeno states at the Zeno time point. In practice, simulating a Zeno hybrid system is challenging in that simulation effectively halts near the Zeno time point. Moreover, due to unavoidable numerical errors, it is not practical to exactly simulate a Zeno hybrid system. Therefore, we propose a method for constructing approximations of Zeno models by leveraging the completed hybrid system model. Using these approximations, we can simulate a Zeno hybrid system model beyond its Zeno point and reveal the complete dynamics of the system being modeled.

[92] A. D. Ames, P. Tabuada and S. Sastry, "On the Stability of Zeno Equilibria," 34-48, Lecture Notes in Com, 3927, Springer-Verlag, 2006.

Zeno behaviors are one of the (perhaps unintended) features of many hybrid models of physical systems. They have no counterpart in traditional dynamical systems or automata theory and yet they have remained relatively unexplored over the years. In this paper we address the stability properties of a class of Zeno equilibria, and we introduce a necessary

paradigm shift in the study of hybrid stability. Motivated by the peculiarities of Zeno equilibria, we consider a form of asymptotic stability that is global in the continuous state, but local in the discrete state. We provide sufficient conditions for stability of these equilibria, resulting in sufficient conditions for the existence of Zeno behavior.

3. Outreach

3.1. Project Training and Development

We continue to use the CHESS Software Lab, which is focused on supporting the creation of publication-quality software in support of embedded systems design. The lab is a room with wireless and wired network connections, a large table for collaborative work, a large format printer (used for UML diagrams and poster preparation), comfortable furniture supporting extended hours of collaborative work, a coffee machine, and a library that inherited a collection of software technology books from the Ptolemy Project. This room is used to promote a local version of the Extreme Programming (XP) software design practice, which advocates pair programming, design reviews, code reviews, extensive use of automated regression tests, and a collaboratively maintained body of code (we use CVS). The room began operation in March of 2003 and has been in nearly constant use for collaborative design work. The principal focus of that work has been on advanced tool architectures for hybrid and embedded software systems design.

3.2. Outreach Activities

Continuing in our mission to build a modern systems science (MSS) with profound implications on the nature and scope of computer science and engineering research, the structure of computer science and electrical engineering curricula, and future industrial practice. This new systems science must pervade engineering education throughout the undergraduate and graduate levels. Embedded software and systems represent a major departure from the current, separated structure of computer science (CS), computer engineering (CE), and electrical engineering (EE). In fact, the new, emerging systems science reintegrates information and physical sciences. The impact of this change on teaching is profound, and cannot be confined to graduate level.

This year we have continued our work to lay the foundation for a new philosophy of undergraduate teaching at the participating institutions. We also used the summer months to foster appreciation for research in underprivileged and minority students in engineering, by continuing to sponsor and participate in the established REU programs SUPERB-IT at UCB and SIPHER at VU.

We continue the collaboration with San Jose State University to continue to develop the undergraduate embedded control course jointly between Berkeley, Vanderbilt and San Jose State University. Prof. Ping Hsu from San Jose State University teaches the class at both Berkeley and San Jose State.

3.2.1. Curriculum Development for Modern Systems Science (MSS)

Our agenda is to restructure computer science and electrical engineering curricula to adapt to a tighter integration of computational and physical systems. Embedded software and systems represent a major departure from the current, separated structure of computer science (CS), computer engineering (CE), and electrical engineering (EE). In fact, the new, emerging systems

science reintegrates information and physical sciences. The impact of this change on teaching is profound, and cannot be confined to graduate level. Based on the ongoing, groundbreaking effort at UCB, we are engaged in retooling undergraduate teaching at the participating institutions, and making the results widely available to encourage critical discussion and facilitate adoption.

We are engaged in an effort at UCB to restructure the undergraduate systems curriculum (which includes courses in signals and systems, communications, signal processing, control systems, image processing, and random processes). The traditional curriculum in these areas is mature and established, so making changes is challenging. We are at the stage of attempting to build faculty consensus for an approach that shortens the pre-requisite chain and allows for introduction of new courses in hybrid systems and embedded software systems.

Undergrad Course Insertion and Transfer

At many institutions, introductory courses are quite large. This makes conducting such a course a substantial undertaking. In particular, the newness of the subject means that there are relatively few available homework and lab exercises and exam questions. To facilitate use of this approach by other instructors, we have engaged technical staff to build web infrastructure supporting such courses. We have built an instructor forum that enables submission and selection of problems from the text and from a library of submitted problems and exercises. A server-side infrastructure generates PDF files for problem sets and solution sets.

The tight integration of computational and physical topics offers opportunities for leveraging technology to illustrate fundamental concepts. We have developed a suite of web pages with applets that use sound, images, and graphs interactively. Our staff has extended and upgraded these applets and created a suite of PowerPoint slides for use by instructors.

We have begun to define an upper division course in embedded software (aimed at juniors and seniors). This new course will replace the control course at the upper division level at San Jose State. We also continued to teach at UC Berkeley the integrated course designed by Prof. Lee, which employs techniques discovered in the hybrid and embedded systems research to interpret traditional signals.

Course: Structure and Interpretation of Signals and Systems (UCB, EECS 20N)
http://ptolemy.eecs.berkeley.edu/eecs20/
Instructor: Prof. Edward A. Lee
Prof. Pravin Varaiya
Prof. Babak Ayazifar

> This course is an introduction to mathematical modeling techniques used in the design of electronic systems. Signals are defined as functions on a set. Examples include continuous time signals (audio, radio, voltages), discrete time signals (digital audio, synchronous circuits), images (discrete and continuous), discrete event signals, and sequences. Systems are defined as mappings on signals. The notion of state is discussed in a general way. Feedback systems and automata illustrate alternative approaches to modeling state in systems. Automata theory is studied using Mealy machines with input and output. Notions of equivalence of automata and concurrent composition are introduced. Hybrid systems combine time-based signals with event sequences. Difference and differential equations are considered as models for linear, time-invariant state

machines. Frequency domain models for signals and frequency response for systems are investigated. Sampling of continuous signals is discussed to relate continuous time and discrete time signals.

Graduate Courses

Several graduate courses were taught in the area of embedded and hybrid systems, as well as systems modeling. All of these courses are a reflection of the teaching and curriculum goals of the ITR and its affiliated faculty.

Course: Autonomous Systems: Algorithms and Implementation (UCB, EECS 290n)
Instructor: Dr. Jonathan Sprinkle,
Prof. S. Shankar Sastry

Autonomous systems can be thought of as robust control systems that can use discrete and continuous control inputs to choose a locally or globally optimal future state. To say that an autonomous system is intelligent is considered redundant by many researchers; though the distinction can be made that intelligent autonomous systems should react in an intuitive manner to stimuli. This course approached the domain of autonomous systems from the perspective of algorithms and their implementation upon an existing software and hardware infrastructure. Our "driving" example will in fact be the newest DARPA Grand Challenge instantiation: the DARPA Urban Challenge.

Course: Embedded System Design: Models, Validation, and Synthesis (UCB EE249)
Instructor: Prof. Alberto Sangiovanni-Vincentelli

This course is about the design of embedded real-time systems. Embedded real-time systems are pervasive in today's world. The methodology used for the design of these devices is still based on principles and tools that are not adequate for the complexity of the applications being developed today. The most important characteristic of these systems is the massive use of programmable components to achieve the design goals. Today, the dominant part of the design effort for embedded system is software. Real-time and power dissipation constraints make embedded software design particularly difficult since traditional abstraction for software do not include physical quantities. The choice of the architecture of the implementation is another essential characteristic of embedded system design. The implementation platform should be selected to support the application of interest optimizing a set of conflicting criteria that include flexibility, scalability, design time, manufacturing cost and reliability. In this course, we will present the principles of a methodology that favors design re-use, formal verification, software design and optimized architecture selection. The basic tenet of the methodology is orthogonalization of concerns, and, in particular, separation of function and architecture, computation and communication. This methodology called platform-based design will be presented as a paradigm that incorporates these principles and spans the entire design process, from system-level specification to detailed circuit implementation.

Course: Foundations of Hybrid and Embedded Systems (VU, CS 376)
Instructor: Prof. Xenofon Koutsoukos

Prof. T. John Koo

Modeling, analysis, and design of hybrid and embedded systems. Heterogeneous modeling and design of embedded systems using formal models of computation, modeling and simulation of hybrid systems, properties of hybrid systems, analysis methods based on abstractions, reachability, and verification of hybrid systems.

Course: Model Integrated Computing (VU, CS 388 / EE 395)
Instructor: Prof. Janos Sztipanovits

Model-Integrated Computing (MIC) addresses the problems of designing, creating, and evolving information systems by providing rich, domain-specific modeling environments including model analysis and model-based program synthesis tools. MIC is used to create and evolve integrated, multiple-aspect models using concepts, relations, and model composition principles routinely used in the specific field, to facilitate systems/software engineering analysis of the models, and to automatically synthesize applications from the models.

Course: Real-Time Systems (VU, EECE 353-01)
Instructor: Prof. Aniruddha Gokhālé
Prof. Douglas Schmidt
Bala Natarajan

This course focuses on the analysis and design of real-time systems. The course covers topics on system modeling using the tagged signal models and timed models of computation, specifications and scheduling techniques for real-time tasks, simulation and verification of real-time systems, software architecture and language for constructing real-time systems. Special attention is paid to computational and simulation tools for real-time systems. Applications ranging from robotics, embedded control systems, drive-by-wire systems, space missions, telecommunication systems, industrial automation, and middleware software systems will be covered.

Course: Automated Verification (VU, EECE 315)
Instructor: Dr. Sherif Abdelwahed

Several notations and methods have been developed to help the designer specify clear and unambiguous system requirements, verify that the requirements are consistent and correct, and verify that the refined design meets its specification. However, these methods are time-consuming and error-prone, and can be applied more effectively if there are tools to check their correctness. The goal of the course is to emphasize formal notations and methods that have tool support. We will cover the basis of underlying theory for the tools.

Course: Automated Verification (VU, EECE 375)
Instructor: Dr. Sherif Abdelwahed

This course provides a detailed coverage of the diagnosis and supervisory control problem for discrete, asynchronous, nondeterministic systems like manufacturing, traffic and communication systems. The underlying theory is developed in an elementary

framework of automata and formal languages, and is supported by a software package for creating applications.

3.2.2. SUPERB-IT Program

SUPERB-IT—2006

The Center for Hybrid and Embedded Software Systems is proud to sponsor four undergraduate students from diverse backgrounds and cultures to participate in SUPERB-IT. These students will interact with individual mentors throughout the summer, and perform research and supporting activities in the area of hybrid and embedded systems. This year, all participants were women engineers.

The Summer Undergraduate Program in Engineering Research at Berkeley–Information Technology (SUPERB–IT) in the Electrical Engineering and Computer Sciences (EECS) Department offers a group of talented undergraduate engineering students the opportunity to gain research experience. The program's objective is to provide research opportunities in engineering to students who have been historically underrepresented in the field for reasons of social, cultural, educational or economic barriers, by affirming students' motivation for graduate study and strengthening their qualifications.

SUPERB-IT participants spent eight weeks at UC Berkeley during the summer of 2006 working on exciting ongoing research projects in information technology with EECS faculty mentors and graduate students. Students who participate in this research apprenticeship explore options for graduate study, gain exposure to a large research-oriented department, and are motivated to pursue graduate study. Additional information about the program can be obtained at:

http://www.eecs.berkeley.edu/Programs/ugrad/superb/superb.html

This ITR project contributed to the support of four SUPERB-IT students in 2006, and organized projects (described below in the abstracts from their papers) in hybrid systems theory, wireless sensor networks, dynamical systems simulation, and autonomous systems. The students were hosted by the Chess center at Berkeley (Center for Hybrid and Embedded Software Systems).

SUPERB-IT participants received a $3,500 stipend, room and board on campus in the International House, and up to $600 for travel expenses. In addition, Chess provided these students with one laptop computer each, configured with appropriate software, plus laboratory facilities for construction of associated hardware.

The students supported at Berkeley in 2006 were Dominique Duncan, Nandita Mitra, Nashlie Sephus, and Heather Taylor. The students worked on individual projects which were tailored to fit their individuation needs, interests, and background, as well as to contribute to the overall goals of the ITR project. Four graduate student mentors facilitated the process, and the operation was coordinated and directed by Professor Shankar Sastry and Dr. Jonathan Sprinkle. Each student was responsible for an individual project, as described below. Their project posters and reports are available at

http://chess.eecs.berkeley.edu/projects/ITR/2006/superb/index.html

In addition to the science of research, we will also expose students to the reporting aspects of research. These include the techniques used for writing papers, technical skills such as the use of LaTeX, and the importance of having a stock version of what exactly it is you are working on.

Specific cross-cutting tasks include LaTeX template design, poster design and best practices, using the Concurrent Versioning System (CVS), participation in a literature reading group, and research reporting through the web.

Project: Highway Traffic Flow Analysis and Control
Student: Dominique Duncan—*University of Chicago*
Mentor: Alexandr Kurzhanskiy

Significant inspiration of computer network topology and communication comes from an empirical understanding of how road networks function. This project takes foundational work by CHESS researchers in hybrid systems to investigate a macroscopic switching-mode model (SMM) of traffic. The goal is to learn how hybrid systems are used for traffic modeling, experiment with different techniques for reachability analysis, implement a controller for the system, and then study the system behavior in the presence of disturbances. The end result will be MATLAB simulations which show the impact of the work.

Dominique Duncan is from Kansas and now a rising senior at the University of Chicago, majoring in Mathematics and Polish Literature. She is working this summer with Mentor Alex Kurzhanskiy.

Her areas of interest are highway traffic control, stochastic systems, and biomedical applications of math and statistics. Her SUPERB work is the flow analysis and control of highway traffic. Dominique plans on pursuing a PhD in electrical engineering and will be applying to schools in Fall 2006.

She is Vice President of the University of Chicago Math Club, President of the UofC Polish American Student Association, member of Women in Science, and member of IEEE CSS Women in Control group.

Project: Tool for probabilistic safety verification of stochastic hybrid systems
Student: Nandita Andromeda Mitra—*Rutgers, the State University of New Jersey*
Mentor: Saurabh Amin and Alessandro Abate

Many safety critical systems like air traffic control involve modeling their behavior as hybrid systems. The effect of uncertain system dynamics and external inputs can be incorporated by modeling the system as a controlled stochastic hybrid system (SHS). Design of controllers for SHS that guarantees a certain safety criterion can be posed as a quantitative verification problem. The goal of this project is to develop a computational tool for stochastic reachability analysis of a benchmark SHS.

Nandita is a rising senior at Rutgers, the State University of New Jersey, majoring in Electrical Engineering. She is working with Saurabh Amin and Alessandro Abate and her project is making a toolbox for probabilistic safety verification of stochastic hybrid systems.

Nandita was born in Hawaii and went to Bangladesh with her parents. After studying two years in Bangladesh University of Engineering and Technology she transferred credit to Rutgers and started as a junior there from Fall 2005

She is the treasurer for IEEE, Rutgers for the academic year of 2006-2007.

Nandita is interested in wireless communication and has become interested in stochastic hybrid systems after coming to Berkeley. Her future plan is to continue working in these fields.

Project: Autopilot for an Ultra-Light, Flying Wing
Student: Nashlie H. Sephus—*Mississippi State University*
Mentor: Todd Templeton

The purpose of this project is to develop an auto-pilot for a small, light fixed-wing aircraft named the Zagi. This aircraft is interesting because it is inexpensive, simple and fast to deploy, and is virtually indestructible since it is made of expanded polypropylene (EPP) foam. This aircraft is also challenging from a control perspective because it is vulnerable to wind and can only carry a minimal payload. First, we develop optimal local trajectories given current wind conditions. These local trajectories are used to determine the path that the vehicle should try to maintain between widely-spaced waypoints, and should trade off between overshooting corners and maintaining the desired trajectory. The testing and results of this portion are performed in MATLAB in order to plot and view the optimizing functionality. We then implement these local trajectories in the Zagi autopilot (written in C) to enable it to follow an incrementally-specified global trajectory with future planning. Initial tests are performed using the CRRCsim simulator interfaced to the Zagi hardware. Results from these tests prove that the planning autopilot is better than the existing autopilot because it directs a path along three points at a time versus only one point. Also, this higher level planner is greedy in that it reaches a desired optimal trajectory without using extra minimizing functionality. Future work involves final testing on a real Zagi at Richmond Field Station.

Nashlie Sephus is a rising senior at Mississippi State University, majoring in Computer Engineering. She is working this summer on an autopilot for a crossbow aircraft with mentor Todd Templeton.

Nashlie's research interests include embedded systems, robotics, and high performance computing. She has also completed three semesters of Cooperative Education in the Information Technology field.

After graduation, she plans to enter graduate school. Her hobbies and other interests include music, tennis, golf, pool, and shopping.

Project: Multihop Routing Simulation of TinyOS-Based Wireless Sensor Networks in Viptos
Student: Heather Taylor—*University of Vermont*
Mentor: Elaine Cheong

Wireless Sensor Networks are a burgeoning area of research and applications in embedded systems. The purpose of this project is to understand and further develop Viptos (Visual Ptolemy and TinyOS), an integrated graphical development and simulation environment for TinyOS-based wireless sensor networks. TinyOS is the operating system for the Berkeley Motes, which are small embedded systems capable of

collecting audio, temperature and other kinds of sensor data and transmitting it via a radio. A TinyOS simulator called TOSSIM is used, a key piece of which is the ability to simulate a network topology once it is functioning. Viptos extends the capabilities of the TinyOS simulator to allow simulation of heterogeneous networks. The final goal of this research is to create a graphical representation of the communication between motes in a multihop network simulation in Viptos. This will be done by adding an entity to Viptos which will collect transmitted information and display a graphical representation of that communication between nodes.

Heather Taylor is a rising senior studying Electrical Computer Engineering at the University of Vermont. This summer she is working with her mentor Elaine Cheong.

Her areas of interest include Wireless Sensor Networks, RFID technology, and embedded systems. This summer she will be working on multihop applications in Viptos.

Her involvement with UVM chapters include: Chair, IEEE; Webmaster, SWE; McNair Scholars, URECA!, & Tau Beta Pi. She plans to pursue her PhD after graduation.

3.2.3. Summer Internship Program in Hybrid and Embedded Software Research (SIPHER) Program

The SIPHER program (Summer Internship Program in Hybrid and Embedded Software Research) is a program similar to SUPERB-IT, but located at Vanderbilt. More information about the program can be found at:

http://fountain.isis.vanderbilt.edu/teaching

The Institute for Software-Integrated Systems (ISIS) at Vanderbilt University's School of Engineering in cooperation with UC Berkeley and University of Memphis has recently been awarded a grant by the National Science Foundation to conduct research in the field of Hybrid and Embedded Systems (HES). The research aims at laying the scientific and technological foundations of embedded system design. Embedded computing systems are present in all traits of modern society: in cars and airplanes, in cell phones, in household devices, in medical devices, just to name a few. This is a multi-year research project that builds the science: the principles and the math, and the technology: the tools that the next generation of engineers will use to build these systems in the future, better than ever before.

The objective of the SIPHER program is that undergraduates from underrepresented groups participate in the research program: receive training in the science and technology developed by the researchers, and work on specific research problems. The program will be coordinated with UC Berkeley's SUPERB-IT, and joint teleconferences are expected.

The undergraduate students who apply to this program are expected to be rising juniors and seniors in an Electrical or Computer Engineering or Computer Science program leading towards a BSc or BE degree. The students must have a background in elementary systems courses: signals and systems for EE, digital systems in CE/CS, and have skills in programming using a high-level language. Engineering students from other programs, like Mechanical Engineering and Chemical Engineering are also encouraged to apply, provided they have similar backgrounds (e.g. in controls).

The SIPHER program runs for 10 weeks each summer, and there are 7 (seven) positions available, with a $6,000 stipend for the period. This year, the participants will be partly funded by the Tennessee Louis Stokes Alliance for Minority Participation (TLSAMP) project (supported by NSF). Students are expected to pay for their accommodations. Limited housing opportunities may be available on the Vanderbilt campus. Applicants are competitively selected to the program.

In the SIPHER activities, we organized a summer internship in 2006 for eight participants from underrepresented groups. The students are organized into groups who solved different embedded software development problems. The students used Vanderbilt-developed modeling tool to create models of the embedded applications, develop code for the components, and then use the model transformation tools to create the final application.

The students worked on small, team-oriented projects related to development of embedded software. In this work they used software tools available at ISIS, and they were supervised by professors and senior graduate students. During first few weeks they underwent rigorous training to learn how to use the design tools. The training was provided by lecturers who deliver our Model-Integrated Computing classes. All of the students had backgrounds in programming, and thus were able to solve the project problems. Similarly to UCB, graduate student mentors assisted and guided the student projects. Descriptions for the projects and the students who worked on them are included below.

In 2006 we had 7 undergraduates supported by the ITR SIPHER program, and one additional undergraduate was supported by an NSF REU grant. All these students worked on small research projects related to the goals of the ITR. The projects are listed below.

Project: Radio Controlled Car Controller
Undergraduates: Jessica Kane, Thao Nguyen
Graduate Student Mentor: Graham Hemingway

> Overview: Using the existing infrastructure for autonomous helicopter flight, students have equipped a small radio-controlled car with sensors so that it can localize itself, then wrote a controller in Simulink to have the car do things such as follow a line autonomously.

Project: Hybrid System Modeling / Fault Diagnosis
Undergraduates: Nathaniel Allotey, Brian Turnbull
Graduate Student Mentor: Wu Jian

> Overview: Students used Simulink/Stateflow to build a model of the three tank system, then designed Stateflow controllers and ran those on the actual system, and also performed diagnosis experiments. Also experimented with adaptive control.

Project: Controlling Lego Robots Using a Synchronous-Reactive Model of Computation
Undergraduates: Javier Lara, Darren White
Graduate Student Mentor: Ethan Jackson

> Overview: Using the synchronous-reactive model of computation, students have built various controllers for the Lego Mindstorm robots, to follow a geometric pattern on the floor.

Project: Exploring with Lego Robots
Undergraduates: Daniel Limbrick, Emily Sherrill
Graduate Student Mentor: Daniel Balasubramanian

Overview: Students used a Max232 interface to adapt a Lego Mindstorms robot to use bluetooth communication, then using java, had the robot explore a maze, communicate the layout of the maze back to a computer, then built a controller to allow the robot to be driven from the computer screen.

4. Publications and Products

In this section, we list published papers only. Submitted papers and in press papers are described in Section 2.2.

4.1. Journal Publications

- Arkadeb Ghosal, Sri Kanajan, Randall Urbance, Alberto Sangiovanni-Vincentelli, *SAE 2007 Transactions: Journal of Passenger Cars: Electronic and Electrical Systems*, "An Initial Study on Monetary Cost Evaluation for the Design of Automotive Electrical Architectures," pp. 844-856, March 2007; ISBN: 978-0-7680-1839-4.

- Alberto L. Sangiovanni-Vincentelli, *Proceedings of the IEEE*, "Quo Vadis SLD: Reasoning about Trends and Challenges of System-Level Design," 95(3):467-506, March 2007.

- Alex A. Kurzhanskiy, Pravin Varaiya, *IEEE Transactions Automatic Control*, "Ellipsoidal Techniques for Reachability Analysis of Discrete-Time Linear Systems," 52(1):26-38, January 2007.

- A. Agrawal, Gabor Karsai, Sandeep Neema, Feng Shi, Attila Vizhanyo, *Journal on Software and System Modeling*, "The Design of a Language for Model Transformations," 5(3):261-288, September 2006.

- Sztipanovits, J., Bay, J., Rohrbough, L., Sastry, S., Schmidt, D., Whitaker, N., Winter, D., *IEEE Computer*, "ESCHER: A New Technology Transitioning Model," 40(3):90-92, March 2007.

4.2. Conference Papers

- Aaron D. Ames, Robert D. Gregg, Haiyang Zheng, Prof. Shankar Sastry, "Is There Life After Zeno? Taking Executions past the Breaking (Zeno) Point," 2007 American Control Conference, ACS, June, 2006; Presented in conjunction with BEARS 2006.

- Bor-Yuh Evan Chang, Matthew Harren, and George C. Necula, "Analysis of Low-Level Code Using Cooperating Decompilers," The 13th International Static Analysis Symposium (SAS), Kwangkeun Yi (ed.), 318-335, September, 2006.

- Ye Zhou and Edward A. Lee, "A Causality Interface for Deadlock Analysis in Dataflow," Proceedings of the Sixth ACM International Conference on Embedded Software (EMSOFT'06), Sang Lyul Min, Wang Yi (eds.), ACM, 44-52, October, 2006.

- Pannag R Sanketi, J. Carlos Zavala, J. K. Hedrick, M. Wilcutts, T. Kaga, "A Simplified Catalytic Converter Model for Automotive Coldstart Applications with Adaptive Parameter Fitting," 8th International Symposium on Advanced Vehicle Control, August, 2006.

- Shinjiro Kakita, Yosinori Watanabe, Douglas Densmore, Abhijit Davare, Alberto Sangiovanni-Vincentelli, "Functional Model Exploration for Multimedia Applications via Algebraic Operators," ACSD 2006 - Sixth International Conference on Application of Concurrency to System Design, 229-238, June, 2006.

- Qi Zhu, Abhijit Davare and Alberto Sangiovanni-Vincentelli, "A Semantic-Driven Synthesis Flow for Platform-Based Design," submitted to Fourth ACM-IEEE International Conference on Formal Methods and Models for Codesign (MEMOCODE'06), July, 2006.

- Abhijit Davare, Jike Chong, Qi Zhu, Douglas Densmore and Alberto Sangiovanni-Vincentelli, "An Overlap-based MILP Formulation for Task Allocation and Scheduling," submitted, October, 2006.

- A. D. Ames and S. Sastry, "Hybrid Geometric Reduction of Hybrid Systems," 2006 45th IEEE Conference on Decision and Control, 923-929, December, 2006.

- A. D. Ames, R. D. Gregg, E. D. B. Wendel and S. Sastry, "Towards the Geometric Reduction of Controlled Three-Dimensional Robotic Bipedal Walkers," Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, July, 2006.

- A. D. Ames and S. Sastry, "Hybrid Routhian Reduction of Lagrangian Hybrid Systems," American Control Conference, June, 2006.

- A. D. Ames and S. Sastry, "Hybrid Cotangent Bundle Reduction of Simple Hybrid Mechanical Systems with Symmetry," American Control Conference, June, 2006.

- A. D. Ames, H. Zheng, R. D. Gregg and S. Sastry, "Is there Life after Zeno? Taking Executions Past the Breaking (Zeno) Point," American Control Conference, June, 2006.

- Antoon Goderisa, Christopher Brooks, lkay Altintas, Edward A. Lee, "Composing Different Models of Computation in Ptolemy II and Kepler," 2007 Proceedings, International Conference on Computational Science (ICCS), May, 2007; To appear at International Conference on Computational Science (ICCS) 2007.

- Takashi Nagata, Hwan Hur, Masayoshi Tomizuka, "Model-Based Control for Smooth Gear Shifting by Engine-AT Collaboration," 8th International Symposium on Advanced Vehicle Control (AVEC '06), 635-640, August, 2006.

- A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry, "Probabilistic Reachability for Safety and Regulation of Controlled Discrete-Time Stochastic Hybrid Systems," Proceedings of the 45th IEEE Conference on Decision and Control, IEEE, December, 2006.

- A. Abate, A. Ames and S. Sastry, "Error Bounds Based Stochastic Approximations and Simulations of Hybrid Dynamical Systems," Proceedings of the 25th American Control Conference, June, 2006.

- A. Abate, A. Ames and S. Sastry, "A-Priori Detection of Zeno Behavior in Communication Networks Modeled as Hybrid Systems," Proceedings of the 25th American Control Conference, June, 2006.

- A. Abate and A. Tiwari, "Box Invariance of hybrid and switched systems," Proceedings of the 2nd IFAC Conference on Analysis and Design of Hybrid Systems, IFAC, June, 2006.

- A. Abate, M. Chen and S. Sastry, "Analysis of an Implementable Application Layer Scheme for Flow Control over Wireless Networks," Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, July, 2006.

- Arkadeb Ghosal, Thomas A. Henzinger, Daniel Iercan, Christoph Kirsch, Alberto Sangiovanni-Vincentelli, "A Hierarchical Coordination Language for Interacting Real-Time Tasks," Proceedings of the Sixth ACM International Conference on Embedded Software (EMSOFT'06), Sang Lyul Min, Wang Yi (eds.), ACM, 132-141, October, 2006.

- Arkadeb Ghosal, Sri Kanajan, Randall Urbance, Alberto Sangiovanni-Vincentelli, "An Initial Study on Monetary Cost Evaluation for the Design of Automotive Electrical Architectures," SAE, April, 2007.

- Nadathur Satish, Kaushik Ravindran and Kurt Keutzer, "A Decomposition-based Constraint Optimization Approach for Statically Scheduling Task Graphs with Communication Delays to Multiprocessors," 10th Conference of Design, Automation and Test in Europe (DATE-07), 230-235, April, 2007.

- Abhijit Davare, Douglas Densmore, Trevor Meyerowitz, Alessandro Pinto, Alberto Sangiovanni-Vincentelli, Guang Yang, Haibo Zeng, Qi Zhu, "A Next-Generation Design Framework for Platform-based Design," DVCon 2007, February, 2007.

- Mark L. McKelvin, Jr., Claudio Pinello, Sri Kanajan, Joseph Wysocki, Alberto Sangiovanni-Vincentelli, "Model-Based Design of Heterogeneous Systems for Fault Tree Analysis," 24th International System Safety Conference, Rodney J. Simmons, Ph.D., Norman J. Gauthier (eds.), System Safety Society, 400-409, August, 2006.

- Krishnendu Chatterjee and Thomas A. Henzinger, "Strategy Improvement for Stochastic Rabin and Streett Games," CONCUR 2006 - Concurrency Theory, 17th International Conference, Christel Baier and Holger Hermanns (eds.), 375-389, August, 2006.

- Krishnendu Chatterjee, Luca de Alfaro, Marco Faella, Thomas A. Henzinger, Rupak Majumdar and Marielle Stoelinga, "Quantitative Compositional Pricing," QEST 06, September, 2006.

- Krishnendu Chatterjee, Luca de Alfaro and Thomas A. Henzinger, "Strategy Improvement for Concurrent Reachability Games," QEST 06, September, 2006.

- Krishnendu Chatterjee, "Concurrent Games with Tail Objectives," CSL 06, September, 2006.

- Krishnendu Chatterjee, "Nash Equilibrium for Upward-Closed Objectives," CSL 06, September, 2006.

- Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger and Jean-Francois Raskin, "Algorithms for Omega-Regular Games with Imperfect Information," CSL 06, September, 2006.

- Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, "Algorithms for Buchi Games," GDV 06, August, 2006.

- Krishnendu Chatterjee and Thomas A. Henzinger, "Assume-guarantee Synthesis," TACAS, March, 2007.

- Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, "Generalized Parity Games," FOSSACS 07, March, 2007.

- Krishnendu Chatterjee, "Optimal Strategy Synthesis for Stochastic Muller Games," FOSSACS 07, March, 2007.

- Thomas Huining Feng, Lynn Wang, Wei Zheng, Sri Kanajan, and Sanjit A. Seshia, "Automatic Model Generation for Black Box Real-Time Systems," Design, Automation and Test in Europe (DATE) Conference, April, 2007.

- Takashi Nagata, Masayoshi Tomizuka, "Engine Torque Control Based on Discrete Event Model and Disturbance Observer," ASME International Mechanical Engineering Congress and Exposition (IMECE2007), (submitted), May, 2007; Draft submitted in May 2007, to be presented in Nov. 2007.

- Sumitra Ganesh, Aaron Ames, Ruzena Bajcsy, "Composition of Dynamical Systems for Estimation of Human Body Dynamics," Proceedings of 10th International Conference on Hybrid Systems Computation and Control 2007, Alberto Bemporad, Antonio Bicchi, Giorgio Buttazzo (eds.), 702-706, April, 2007.

- Thomas A. Henzinger and Vinayak Prabhu, "Timed alternating-time temporal logic," Formal Modeling and Analysis of Timed Systems, 4th International Conference, FORMATS 2006, Paris, France, LCNS 4202, Asarin, Eugene; Bouyer, Patricia (eds.), 1-17, September, 2006.

- Gang Zhou, Man-Kit Leung, and Edward A. Lee, "A Code Generation Framework for Actor-Oriented Models with Partial Evaluation," Proceedings of International Conference on Embedded Software and Systems 2007, LNCS 4523, Y.-H. Lee et al. (ed.), 786-799, May, 2007.

- Xiaojun Liu, Eleftherios Matsikoudis, and Edward A. Lee, "Modeling Timed Concurrent Systems," CONCUR 2006 - Concurrency Theory, 17th International Conference, Christel Baier and Holger Hermanns (eds.), 1-15, August, 2006.

- Adam Cataldo, Edward Lee, Xiaojun Liu, Eleftherios Matsikoudis, and Haiyang Zheng, "A Constructive Fixed-Point Theorem and the Feedback Semantics of Timed Systems," Workshop on Discrete Event Systems, July, 2006.

- Aaron D. Ames and Shankar Sastry, "Hybrid Routhian Reduction of Lagrangian Hybrid Systems," American Control Conference, June, 2006.

- Aaron D. Ames and Shankar Sastry, "Hybrid Cotangent Bundle Reduction of Simple Hybrid Mechanical Systems with Symmetry," American Control Conference, June, 2006.

- Aaron D. Ames, Haiyang Zheng, Robert D. Gregg and Shankar Sastry, "Is there Life after Zeno? Taking Executions Past the Breaking (Zeno) Point," American Control Conference, June, 2006.

- Aaron D. Ames, Robert D. Gregg, Eric D.B. Wendel and Shankar Sastry, "Towards the Geometric Reduction of Controlled Three-Dimensional Bipedal Robotic Walkers," 3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control, July, 2006.

- Aaron D. Ames and Shankar Sastry, "Hybrid Geometric Reduction of Hybrid Systems," 45th IEEE Conference on Decision and Control, 923 - 929, December, 2006.

- Thomas A. Henzinger and Joseph Sifakis, "The embedded systems design challenge," Proceedings of the 14th International Symposium on Formal Methods (FM), Lecture Notes in Computer Science, Springer, August, 2006.

- Thomas A. Henzinger, "Games, time, and probability: Graph models for system design and analysis," Proceedings of the 33rd International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM), Lecture Notes in Computer Science, Springer, January, 2007.

- Wei Zheng, Marco Di Natale, Claudio Pinello, Paolo Giusto, Alberto Sangiovanni-Vincentelli, "Synthesis of task and message activation models in real-time distributed automotive systems," Design, Automation and Test in Europe, April, 2007.

- Marco Di Natale, Wei Zheng, Claudio Pinello, Paolo Giusto, Alberto Sangiovanni-Vincentelli, "Optimizing end-to-end latencies by adaptation of the activation events in distributed automotive systems," Real-Time and Embedded Technology and Applications Symposium, April, 2007.

- D. Balasubramanian, A. Narayanan, S. Neema, F. Shi, R. Thibodeaux, G. Karsai, "A Subgraph Operator for Graph Transformation Languages," Proceedings of 6th International Workshop on Graph Transformation and Visual Modeling Techniques, Braga, Portugal, March, 2007.

- G. Karsai, A. Narayanan, "On the Correctness of Model Transformations in the Development of Embedded Systems," Proceedings of the 2006 Monterey Workshop, October, 2006; (Paper presented at conference and submitted to publisher).

- Isaac Amundson, Manish Kushwaha, Xenofon Koutsoukos, Sandeep Neema, and Janos Sztipanovits, "Efficient Integration of Web Services in Ambient-aware Sensor Network Applications," 3rd IEEE/CreateNet International Workshop on Broadband Advanced Sensor Networks (BaseNets 2006), October, 2006.

- Karsai, G., Ledeczi, A., Neema, S., Sztipanovits, J., "The Model-Integrated Computing Toolsuite: Metaprogrammable Tools for Embedded Control System Design," Proc. of the IEEE Joint Conference CCA, ISIC and CACSD, Munich, Germany, 50-55, October, 2006.

- Jackson, E., Sztipanovits, J., "Towards A Formal Foundation For Domain Specific Modeling Languages," Proceedings of the Sixth ACM International Conference on Embedded Software (EMSOFT'06), Sang Lyul Min, Wang Yi (eds.), ACM, 53-63, October, 2006.

- Emerson M., Sztipanovits J., "Techniques for Metamodel Composition," OOPSLA – 6th Workshop on Domain Specific Modeling, 123-139, October, 2006.

- Emerson M., Duncavage S., Mathe J., Sztipanovits J., "WiNeSim: A Wireless Network Simulation Tool," Proceedings of the Sixth ACM International Conference on Embedded Software (EMSOFT'06), Sang Lyul Min, Wang Yi (eds.), ACM, October, 2006.

- Sztipanovits, J., "Towards the Compositional Specification of Semantics for Heterogeneous Domain-Specific Modeling Languages," Workshop on Foundation of Composition, October, 2006.

- Kai Chen, Janos Sztipanovits, Sandeep Neema, "A Case Study on Semantic Unit Composition," Workshop on Modeling in Software Engineering, ICSE 2007 (MISE 2007), May, 2007.

- Kai Chen, Janos Sztipanovits, Sandeep Neema, "Compositional Specification of Behavioral Semantics," Proceedings of Design Automation and Test in Europe Conference (DATE 07), 906-911, April, 2007.

- Manish Kushwaha, Isaac Amundson, Xenofon Koutsoukos, Sandeep Neema, and Janos Sztipanovits, "OASiS: A Programming Framework for Service-Oriented Sensor Networks," In IEEE/Create-Net COMSWARE 2007, January, 2007.

- Sztipanovits, J., "Model-based Software Development," ESMD-SW Workshop, NASA, March, 2007.

- I. Roychoudhury, M. Daigle, G. Biswas, X. Koutsoukos, and P. J. Mosterman, "A Method for Efficient Simulation of Hybrid Bond Graphs," International Conference on Bond Graph Modeling and Simulation (ICBGM 2007), 177-184, January, 2007.

- M. Daigle, I. Roychoudhury, G. Biswas, and X. Koutsoukos, "Efficient Simulation of Component-Based Hybrid Models Represented as Hybrid Bond Graphs," Hybrid Systems: Computation and Control (HSCC 2007), Lecture Notes in Computer Science, vol. 4416, 680-683, April, 2007.

- Yang Zhao, Jie Liu and Edward A. Lee, "A Programming Model for Time-Synchronized Distributed Real-Time Systems," 13th IEEE Real Time and Embedded Technology and Applications Symposium, 2007. RTAS '07, 259 - 268, April, 2007.

- Edward A. Lee and Stephen Edwards, "Precision Timed (PRET) Computation in Cyber-Physical Systems," National Workshop on High Confidence Software Platforms for Cyber-Physical Systems: Research Needs and Roadmap, November, 2006.

- Edward A. Lee and Yang Zhao, "Reinventing Computing for Real Time," Proceedings of the 2006 Monterey Workshop, October, 2006.

- Yang Zhao, Edward A. Lee and Jie Liu, "Application of Programming Temporally Integrated Distributed Embedded Systems," Proceedings of 2006 IEEE 1588 Conference, October, 2006.

- Edward A. Lee, "Cyber-Physical Systems - Are Computing Foundations Adequate?," Position Paper for NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap, October, 2006.

- Edward A. Lee, "Concurrent Semantics without the Notions of State or State Transitions," Formal Modeling and Analysis of Timed Systems, 4th International Conference, FORMATS 2006, Paris, France, LCNS 4202, Asarin, Eugene; Bouyer, Patricia (eds.), 18-31, September, 2006

4.3. Books, Reports, and Other One-Time Publications

- Haiyang Zheng, Edward A. Lee and Aaron D. Ames, Joao Hespanha, Ashish Tiwari (eds.), "Beyond Zeno: Get on with It!," 568-582, 3927, Springer Berlin/Heidelberg, 2006; From "Hybrid Systems: Computation and Control, Proceedings". 3927, ISBN: 3-540-33170-0.

- A. D. Ames, P. Tabuada and S. Sastry, "On the Stability of Zeno Equilibria," 34-48, Lecture Notes in Com, 3927, Springer-Verlag, 2006.

- Jeff Gray, Juha-Pekka Tolvanen, Steven Kelly, Aniruddha Gokhale, Sandeep Neema, and Jonathan Sprinkle, Paul A. Fishwick (ed.), "Domain-Specific Modeling (in CRC Handbook of Dynamic System Modeling)," 7, (in publication), CRC Press, 2007.

- A. Abate S. Amin and M. Prandini and J. Lygeros and S. Sastry, A. Bemporad A. Bicchi and G. Buttazzo (eds.), "Computational Approaches to Reachability Analysis of Stochastic Hybrid Systems," 4-17, 4416, Springer Verlag, 2007.

- A. Abate, A. D'Innocenzo, G. Pola, M.D. Di Benedetto and S. Sastry, A. Bemporad and A. Bicchi and G. Buttazzo (eds.), "The Concept of Deadlock and Livelock in Hybrid Control Systems," 628-632, 4416, Springer Verlag, 2007.

- S. Amin and A. Abate and M. Prandini and J. Lygeros and S. Sastry, J. Hespanha and A. Tiwari (eds.), "Reachability analysis for controlled discrete time stochastic hybrid systems," 49-63, 3927, Springer Verlag, 2007.

- Aaron D. Ames, "Homotopy Meaningful Hybrid Model Structures," American Mathematical Society, 2007; To appear in "Topology and Robotics", AMS Contemporary Mathematics Series, 2007.

- Matthew Emerson. Sandeep Neema. Janos Sztipanovits; Insup Lee, Joseph Leung, Sang H. Son (eds.), "Handbook of Real-Time and Embedded Systems," CRC Press, 2006; ISBN: 1584886781.

- Alessandro Abate, Ashish Tiwari and S. Shankar Sastry, Technical report, "The concept of Box Invariance for biologically-inspired dynamical systems," University of California, Berkeley, UCB/EECS-2006-185, December, 2006.

- Joseph Gerard Makin and Alessandro Abate, Technical report, "A Neural Hybrid-System Model of the Basal Ganglia," University of California, Berkeley, UCB/EECS-2007-16, January, 2007.

- Allen Y. Yang, John Wright, Shankar Sastry, and Yi Ma, Technical report, "Unsupervised Segmentation of Natural Images via Lossy Data Compression," University of California, Berkeley, UCB/EECS-2006-195, December, 2006.

- Kai Chen, Janos Sztipanovits, Sandeep Neema, Technical report, "Compositional Specification of Behavioral Semantics," Institute for Software Integrated Systems (ISIS), ISIS-06-705, June, 2006.

- Edward A. Lee, Technical report, "Computing Foundations and Practice for Cyber-Physical Systems: A Preliminary Report," University of California, Berkeley, UCB/EECS-2007-72, May, 2007.

- Gang Zhou, Man-Kit Leung and Edward A. Lee, Technical report, "A Code Generation Framework for Actor-Oriented Models with Partial Evaluation," University of California, Berkeley, UCB/EECS-2007-29, February, 2007.

- Yang Zhao, Yuhong Xiong, Edward A. Lee, Xiaojun Liu and Lizhi C. Zhong, Technical report, "The Design and Application of Structured Types in Ptolemy II," University of California, Berkeley, UCB/EECS-2007-21, January, 2007.

- Christopher Brooks, Edward A. Lee, Xiaojun Liu, Stephen Neuendorffer, Yang Zhao and Haiyang Zheng, Technical report, "Heterogeneous Concurrent Modeling and Design in Java (Volume 1: Introduction to Ptolemy II)," University of California, Berkeley, UCB/EECS-2007-7, February, 2007.

- Christopher Brooks, Edward A. Lee, Xiaojun Liu, Stephen Neuendorffer, Yang Zhao and Haiyang Zheng, Technical report, "Heterogeneous Concurrent Modeling and Design in Java (Volume 2: Ptolemy II Software Architecture)," University of California, Berkeley, UCB/EECS-2007-8, February, 2007.

- Christopher Brooks, Edward A. Lee, Xiaojun Liu, Stephen Neuendorffer, Yang Zhao and Haiyang Zheng, Technical report, "Heterogeneous Concurrent Modeling and Design in Java (Volume 3: Ptolemy II Domains)," University of California, Berkeley, UCB/EECS-2007-9, February, 2007.

- Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, Technical report, "Strategy Logic," University of California, Berkeley, UCB/EECS-2007-78, May, 2007.

- Thomas Brihaye, Thomas A. Henzinger, Vinayak Prabhu and Jean-Francois Raskin, Technical report, "Minimum-Time Reachability in Timed Games," University of California, Berkeley, UCB/EECS-2007-47, April, 2007.

- Edward A. Lee, Xiaojun Liu and Stephen Andrew Neuendorffer, Technical report, "Classes and Inheritance in Actor-Oriented Design," University of California, Berkeley, UCB/EECS-2006-154, November, 2006.

- Stephen Edwards and Edward A. Lee, Technical report, "The Case for the Precision Timed (PRET) Machine," University of California, Berkeley, UCB/EECS-2006-149, November, 2006.

- Ye Zhou and Edward A. Lee, Technical report, "Causality Interfaces for Actor Networks," University of California, Berkeley, UCB/EECS-2006-148, November, 2006.

- Krishnendu Chatterjee, Thomas A. Henzinger and Nir Piterman, Technical report, "Generalized Parity Games," University of California, Berkeley, UCB/EECS-2006-144, November, 2006.

- Krishnendu Chatterjee, Rupak Majumdar and Thomas A. Henzinger, Technical report, "Stochastic Limit-Average Games are in EXPTIME," University of California, Berkeley, UCB/EECS-2006-143, November, 2006.

- Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger and Jean-Francois Raskin, Technical report, "Algorithms for Omega-Regular Games with Incomplete Information," University of California, Berkeley, UCB/EECS-2006-89, June, 2006.

- Krishnendu Chatterjee and Thomas A. Henzinger, Technical report, "Reduction of Stochastic Parity to Stochastic Mean-payoff Games," University of California, Berkeley, UCB/EECS-2006-140, November, 2006.

- Abhijit Davare, Jike Chong, Qi Zhu, Douglas Michael Densmore and Alberto L. Sangiovanni-Vincentelli, Technical report, "Classification, Customization, and Characterization: Using MILP for Task Allocation and Scheduling," University of California, Berkeley, UCB/EECS-2006-166, December, 2006.

- Ethan Jackson, Technical report, "The Software Engineering of Domain-Specific Modeling Languages: A Survey Through Examples," Institute For Software Integrated Systems (ISIS), ISIS-07-807, March, 2007.

4.4. Dissemination

Although this is a long term project focused on foundations, we are actively working to set up effective technology transfer mechanisms for dissemination of the research results. A major part of this is expected to occur through the open dissemination of software tools.

4.4.1. Software Maturation

Making these software tools useful and usable outside the research community is a significant issue. Towards this end, we have cooperated with the formation of the Escher consortium, which has begun operating (www.escherinstitute.org). Escher has negotiated with both Berkeley and Vanderbilt specific priorities for "industrial hardening" of research tools from this project. In particular, at Berkeley, top priority will be placed on Giotto, xGiotto, and Ptolemy II in the near term. At Vanderbilt, top priority will be placed on GME, Desert, and GReAT. General Motors, Raytheon, and Boeing are signed up as charter industrial partners in Escher, and more companies are expected.

Industry Technology Transition

1. The MIC tool suite is included in the ESCHER maturation program funded by GM, Boeing and Raytheon. The tool suite is now fully integrated and accessible through the quality controlled ESCHER Repository.
2. The MIC tool suite has been adopted by the Boeing Company, Lead System Integrator for the Future Combat Systems (FCS) program, for architecture modeling, architecture exploration and model-based system integration. These applications have opened up new dimensions for the industrial applicability of model-based design approaches.
3. GM Research has continued working with the MIC tool suite in a new experimental vehicle program: Smart Adaptive Vehicle (SAV-2). This effort is a significant driver for our semantic foundations work and provides for us a continuous stream of "industrial strength" challenges.
4. The Raytheon Company has completed the development of the Signal Processing Platform tool suite based on MIC and tests its application in various programs.
5. We continue interaction with Microsoft Visual Studio Product Division on using our experience in the Software Factory product line, and started working with Microsoft Research on expanding our collaboration on Abstract State Machines, the Abstract State Machine Language (AsmL) and its application in semantic anchoring.

4.4.2. Working Groups and Standards

An important, emerging forum for dissemination of information and influencing industrial practice is the recently form Model-Integrated Computing Special Interest Group (MIC PSIG) of OMG. (http://mic.omg.org/) This forum is run by industry and its primary goal is the preparation and management of standardization activities related to various aspects model-based design in embedded systems. The ISIS and CHESS teams are very much involved these activities. The White Paper for a standard Open Tool Integration Framework (OTIF) is based on the work of researcher at ISIS.

4.4.3. "Foundations" The 2006-2007 Chess seminar series

The Chess seminar series provides a weekly forum for the problems and solutions found and solved by Chess members, as well as ongoing research updates. This forum works best when the audience is diverse in background, because the goal is to aid researchers in seeing how the other sub-disciplines are approaching similar problems, or to encourage them to work on problems they had not yet considered.

A common thread for this year's seminar series is "Foundations", which will examine how the work pursued in hybrid systems and embedded software systems can be used in other domains, as well as understanding how to apply foundational results appropriately. An additional angle in this thread is to present an interesting application which can be subdivided into its component parts, or examined from a theoretical standpoint. We also encourage appropriate ideas outside of this scope, since we advocate this diverse seminar series to stay abreast of current research trends.

A full listing of this project-year's speakers is below. Most talks can be downloaded from the seminar website, at http://chess.eecs.berkeley.edu/seminar.htm

- "A Design Flow for the Development, Characterization, and Refinement of System Level Architectural Services"
  Douglas Densmore, UC Berkeley, May 15, 2007.
- "Automating Sequential Clock Gating for Power Optimization"
  Anmol Mathur, Calypto Design Systems, May 1, 2007.
- "Closed-loop impulse control and the theory of Hybrid Systems"
  Alexander B. Kurzhanski, UC Berkeley, April 24, 2007.
- "Automated Extraction of Inductive Invariants to Aid Model Checking"
  Mike Case, UC Berkeley, April 10, 2007.
- "From Concept to Silicon"
  Vason P. Srini, UC Berkeley, April 3, 2007.
- "SAT Sweeping with Local Observability Don't-Cares"
  Nathan Kitchen, UC Berkeley, March 20, 2007.
- "Selective Term-Level Abstraction Using Type-Inference"
  Bryan Brady, UC Berkeley, March 13, 2007.
- "Models for Data-Flow Sequential Processes"
  Mark B. Josephs, London South Bank University, March 13, 2007.
- "Control of Hybrid Systems: Theory, Computation and Applications"
  Manfred Morari, ETH Zurich, March 6, 2007.
- "Time-Portable Real-Time Programming with Exotasks"
  Christoph Kirsch, University of Salzburg, February 27, 2007.
- "High Level Mathematical Modeling and Parallel/GRID Computing with Modelica using OpenModelica"
  Peter Fritzson, Linköpings Universitet, February 20, 2007.
- "Discrete Event Models: Getting the Semantics Right"
  Edward A. Lee, UC Berkeley, February 6, 2007.
- "SCADA for distributed systems: application to the control of irrigation canals"
  Xavier Litrico, Cemagref, January 30, 2007.
- "Resource-Aware Programming"
  Walid Taha, Rice University, January 23, 2007.
- "A Model-Driven Approach to Embedded Control System Implementation"
  Jan F. Broenink, University of Twente, January 19, 2007.
- "The Power of Higher-Order Components in System Design"
  Adam Cataldo, UC Berkeley, December 12, 2006.

- "Some Results on Optimal Estimation and Control for Lossy Networked Control Systems"
  Luca Schenato, University of Padova, December 5, 2006.
- "Graphical System Design: Bringing Embedded Design to the Masses in Science and Engineering"
  Hugo A. Andrade and Zach Nelson, National Instruments, November 28, 2006.
- "Dynamical constrained Impulse system analysis through viability approach and applications"
  Patrick Saint-Pierre, University Paris Dauphine, November 21, 2006.
- "SHIM: A Scheduling-Independent Concurrent Language for Embedded Systems"
  Stephen A. Edwards, Columbia University, November 8, 2006.
- "Port-based Modeling and Control for Efficient Bipedal Walking Machines"
  Vincent Duindam, UC Berkeley, October 31 and November 7, 2006.
- "Using mathematical modeling to help decode biological circuits"
  Claire Tomlin, UC Berkeley & Stanford, October 17, 2006.
- "Control of Hybrid Systems: a Robust Finite State Machine Approach"
  Danielle C. Tarraf, Caltech, September 29, 2006.
- "Modeling with the Timing Definition Language (TDL)"
  Wolfgang Pree, University of Salzburg, September 26, 2006.
- "Advanced Visual Tracking with Bayesian Filter"
  Li-Chen Fu, NTU Taiwan, August 8, 2006

4.4.4. Workshops and Invited Talks

In addition to the below invited and workshop organizational activities, Chess faculty have delivered numerous plenary talks, invited talks, as well as informal dissemination of Chess goals and research.

6th OOPSLA Workshop on Domain-Specific Modeling

During the OOPSLA Conference in October of 2006, Dr. Jonathan Sprinkle participated in, and helped to organize, the 6th OOPSLA Workshop on Domain-Specific Modeling. Domain-Specific Modeling aims at raising the level of abstraction beyond programming by specifying the solution directly using domain concepts. In a number of cases the final products can be generated from these high-level specifications. This automation is possible because of domain-specificity: both the modeling language and code generators fit to the requirements of a narrow domain only, often in a single company. This is the fifth workshop on Domain-Specific Modeling, following the encouraging experiences from the earlier workshops at past OOPSLA conferences (Tampa 2001, Seattle 2002, Anaheim 2003, Vancouver 2004 and San Diego 2005). During the time the DSM workshops have been organized, interest in domain-specific modeling languages, metamodeling and supporting tools has seen a revival. The electronic version of the proceedings, presentation slides and group work results is available at

http://www.dsmforum.org/events/

2007 International Symposium on Code Generation and Optimization (CGO)
March 11-14, 2007: Professor Edward A. Lee developed a position statement, "Are new languages necessary for multicore?" and presented it a panel by the same name at CGO.

Real-Time and Embedded Technology and Applications Symposium (RTAS)
April 3-6, 2007: Professor Edward A. Lee presented the keynote presentation, "Is Truly Real-Time Computing Becoming Unachievable?," at RTAS in Bellevue, WA on April 3-6, 2007.

6th ACM & IEEE Conference on Embedded Software (EMSOFT'06)
October 22-25, 2006: Professor Tom Henzinger was on the executive committee and chaired the advisory committee. Profs. Edward A. Lee, Alberto Sangiovanni-Vincentelli, and Janos Sztipanovits were members of the advisory committee.

4.4.5. General Dissemination

The Chess website, http://chess.eecs.berkeley.edu, includes publications and software distributions. In addition, as part of the outreach effort, the UC Berkeley introductory signals systems course, which introduces hybrid systems, is available at http://ptolemy.eecs.berkeley.edu/eecs20/ and Ptolemy II software is available at http://ptolemy.eecs.berkeley.edu.

The ISIS website, http://www.isis.vanderbilt.edu, makes publications and software available.

4.5. Other Specific Product

The following software packages have been made available during this review period on the Chess website, http://chess.eecs.berkeley.edu:

- Metropolis consists of an infrastructure, a tool set, and design methodologies for various application domains. The infrastructure provides a mechanism such that heterogeneous components of a system can be represented uniformly and tools for formal methods can be applied naturally. The latest release, Metropolis 1.1.2 was made available on October 12, 2006 and may be found at: http://chess.eecs.berkeley.edu/chess/forum/17.html
- The Generic Modeling Environment (GME) is a configurable toolkit for creating domain-specific modeling and program synthesis environments. The configuration is accomplished through metamodels specifying the modeling paradigm (modeling language) of the application domain. The modeling paradigm contains all the syntactic, semantic, and presentation information regarding the domain; which concepts will be used to construct models, what relationships may exist among those concepts, how the concepts may be organized and viewed by the modeler, and rules governing the construction of models. The modeling paradigm defines the family of models that can be created using the resultant modeling environment. The latest release, GME 6.11.9, was released on December 1, 2006 and may be found at: http://www.isis.vanderbilt.edu/projects/gme/.
- GReAT (Graph Rewriting And Transformation) is a component technology of GME comprised of a metamodel based graph transformation language useful for the specification and implementation of model-to-model transformations. The latest release,

GReAT 1.6.0, was released on December 1, 2006 and may be found at: http://www.escherinstitute.org/Plone/tools/suites/mic/great

- Universal Data Model (UDM) generates C++ API from UML class diagrams. The API can be used to read/write XML files, GME databases, etc. and is component technology for Graph Rewriting And Transformation (GReAT). The most recent release, UDM 3.1.1, was released on December 1, 2006 and may be found at: http://www.escherinstitute.org/Plone/tools/suites/mic/udm

- The Ellipsoidal Toolbox is a standalone set of easy-to-use configurable MATLAB routines to perform operations with ellipsoids and hyperplanes of arbitrary dimensions. It computes the external and internal ellipsoidal approximations of geometric (Minkowski) sums and differences of ellipsoids, intersections of ellipsoids and intersections of ellipsoids with halfspaces and polytopes; distances between ellipsoids, between ellipsoids and hyperplanes, between ellipsoids and polytopes; and projections onto given subspaces. Ellipsoidal methods are used to compute forward and backward reach sets of continuous- and discrete-time piecewise affine systems. Forward and backward reach sets can be also computed for continuous-time piece-wise linear systems with disturbances. It can be verified if computed reach sets intersect with given ellipsoids, hyperplanes, or polytopes. The toolbox provides efficient plotting routines for ellipsoids, hyperplanes and reach sets ET version 1.1 was released on December 10, 2006 and is available at www.eecs.berkeley.edu/~akurzhan/ellipsoids

- Ptplot is a 2D signal plotter implemented in Java. Ptplot can be used in a standalone applet or application or used in your own applet or application. PtPlot 5.6 was released on January 15, 2007 and is available as part of Ptolemy or as a standalone download at http://ptolemy.eecs.berkeley.edu/java/ptplot

- Ptolemy II 6.0.2 is a set of Java packages supporting heterogeneous, concurrent modeling and design. Its kernel package supports clustered hierarchical graphs, which are collections of entities and relations between those entities. Its actor package extends the kernel so that entities have functionality and can communicate via the relations. Its domains extend the actor package by imposing models of computation on the interaction between entities. Examples of models of computation include discrete-event systems, data flow, process networks, synchronous/reactive systems, and communicating sequential processes. Ptolemy II includes a number of support packages, such as data, providing a type system, data encapsulation and an expression parser, plot, providing visual display data, math, providing matrix and vector math and signal processing functions, and graph, providing graph-theoretic manipulations. The Ptolemy II 6.0.2 was released by UC Berkeley on February 4, 2007 and is available at http://ptolemy.eecs.berkeley.edu/ptolemyII/

- CIL is a front-end for the C programming language that facilitates program analysis and transformation. CIL will parse and type check a program, and compile it into a simplified subset of C. For example, in CIL all looping constructs are given a single form and expressions have no side-effects. This reduces the number of cases that must be considered when manipulating a C program. CIL has been used for a variety of projects, including CCured, a tool that makes C programs memory safe. CIL supports ANSI C as

well as most of the extensions of the GNU C and Microsoft C compilers. A Perl script acts as a drop in replacement for either gcc or Microsoft's cl, and allows merging of the source files in your project. Other features include support for control-flow and points-to analyses. CIL Version 1.3.6 was released on February 5, 2007 and is available at http://cil.sourceforge.net.

- Viptos (Visual Ptolemy and TinyOS) is an integrated graphical development and simulation environment for TinyOS-based wireless sensor networks. Viptos allows developers to create block and arrow diagrams to construct TinyOS programs from any standard library of nesC/TinyOS components. The tool automatically transforms the diagram into a nesC program that can be compiled and downloaded from within the graphical environment onto any TinyOS-supported target hardware. In particular, Viptos includes the full capabilities of VisualSense, which can model communication channels, networks, and non-TinyOS nodes. Viptos is compatible with nesC 1.2 and includes tools to harvest existing TinyOS components and applications and convert them into a format that can be displayed as block (and arrow) diagrams and simulated Viptos 1.0.2 was released on February 9, 2007 and is available at http://ptolemy.eecs.berkeley.edu/viptos/
- SKETCH is a sketching system based on combinatorial search, as opposed to transformations. In this system, the sketch is given in the form of a partial program--a program with holes--and the sketch resolution synthesizes code to fill in the holes. The holes may stand for index expressions, lookup tables or bitmasks, and the programmer can easily define new kinds of holes with the synthesis operators provided. SKETCH completes sketches by means of a combinatorial search based on generalized Boolean satisfiability. SKETCH 0.9.5 was released on April 23, 2007 and is available at http://javasketch.sourceforge.net/

5. Contributions

This section summarizes the major contributions during this reporting period.

5.1. Within Discipline

5.1.1. Hybrid Systems Theory

- We have worked with our definition of an operational semantics for hybrid systems in the current and next generation of toolsets to reflect these semantics.
- We have developed algorithms for computing the real value of discounted properties, and continued investigation of their application.
- We have matured a theory of a homology theory of hybrid systems which enables elegant characterization of Zeno and other qualitative properties of hybrid systems.
- We have improved on the best known algorithms for finding strategies for the control of stochastic hybrid systems.
- We have continued development of a toolbox using ellipsoidal methods to calculate reach sets for linear dynamic systems, and begun to apply those to hybrid systems.
- We have developed an extensive theory of two and multi person stochastic games with extensions of notions of safety and almost safety in a number of important directions.
- We have continued to apply and study stochastic hybrid systems within the domain of biological systems.
- We are developing a static analysis mechanism that infers the common causality properties of a modal model from those of its modes. The result of the static analysis is conservative, but provides safety guarantees.
- We have continued in our broad initiative to support tool chains in hybrid systems under semantic anchoring and model transformations.
- We derived verifiable necessary and sufficient conditions on when composition preserves semantics for a heterogeneous network of embedded systems.
- We have formally proved the benefits of the logical execution time (LET) model in terms of composability over traditional real-time models.
- We have developed a technique to extend the simulation of a hybrid system past its Zeno point, reducing the computational burden past that point and revealing the complete behavior of the system.

5.1.2. Model-Based Design

- We have developed the first release of a semantic anchoring tool suite, and have demonstrated the use of the tool infrastructure in specifying the semantics of hierarchical state automata.

- Using various specifications of timed automata, we have examined approaches for defining semantic units. We demonstrated the concepts with developing a semantic unit for timed automata and showed the anchoring of UPAAL and IF to this common semantic unit.

- We started investigating the problems of defining semantics for heterogeneous modeling languages, and began establishing a composition theory for semantic units.

- Applying our ongoing work on metamodeling, we have continued development on semantic anchoring for model-based development. Specifically, we have extended the semantic anchoring framework to heterogeneous behaviors.

- We have continued to demonstrate our defined agent algebras as a formal framework for uniformly representing and reasoning about models of computation used in the design of hybrid and embedded software systems.

- We have continued to demonstrate our theoretical and compositional framework for reasoning about causality in components which are composed under concurrent models of computation.

- We have extended our previously developed tagged-signal model for concurrent models of computation to represent the semantics of globally asynchronous, locally synchronous systems built upon loosely time-triggered architectures.

- We have continued to maintain a language and a suite of supporting tools for the specification of model transformations based on graph rewriting.

- We have continued to use our approach to model synthesis based on patterns specified formally as metamodels.

- We have developed an interface theory based approach to static analysis of actor models through composition. It results in an automaton which will contain information used for further static analysis of a composed actor model.

- We have developed a new component model for timed models of computation such as discrete event, continuous time, hybrid systems, and synchronous/reactive models.

- We have built a scalable and formal specification language for embedded systems which can use constraint checking to auto-generate parts of a specification and to approximate the correctness of the specification without invoking verification tools

5.1.3. Advanced Tool Architectures

- We have further developed the code generation approach based on component specialization by developing a formal framework for reasoning about reconfiguration in embedded software.

- We have continued to improve the performance and feature set of the Metropolis framework.

- We have further developed our notion of interface theories to support reasoning about heterogeneous component composition and about the dynamics of models of computation.

- We formulated and solved the task allocation problem for a popular multithreaded, multiprocessor embedded system, the Intel IXP1200 network processor.

- We have continued to investigate interests in fault-tolerant systems by developing new modeling languages which simulate and trace faults in a system.

- We have continued development of the Ptolemy II tool suite, including HyVisual, VisualSense, and Viptos tools for hybrid systems, sensor networks, and NesC-based wireless sensor programming.

- We have shown how to guarantee type-safety in legacy C programs and verify memory safety in the assembly code.

- We have strengthened our understanding of discounted reward objectives to yield real-numbered quantities (e.g., power consumption) that can be expressed during verification.

5.1.4. Experimental Research

- We have extended model predictive control for hybrid systems with a finite control set to develop air and water recovery systems for the NASA Advanced Life Support (ALS) system for long-duration missions.

- We have begun to apply our previous work on safe set calculations to the Autonomous Aerial Refueling (AAR) while in formation problem.

- We have deployed the Metropolis platform-based design methodology for use on various veitronics problems of interest to Toyota, GM, and BMW.

- We have continued development, and deployed a modeling environment for wireless sensor networks. These have been used to simulate detection of a dirty bomb.

- We have developed new programming models for sensor networks that build on the popular TinyOS models.

- We have shown how compositional technologies can be used to produce an autonomous helicopter in the loop with a camera to choose a landing zone, and physically land the vehicle.

- We have used reachability to perform analysis of the cold start problem and shown anticipated reduction in raw hydrocarbon emissions during warm-up using a hybrid systems model.

- We have shown how fault tolerant data flow can be used to synthesize real-time feedback controllers for safety critical applications.

- We have shown that hybrid systems theory can be coupled with Lagrangian methods to produce reduced state-space expressions of computationally difficult problems, such as the motion of a bipedal walker.

5.2. Other Disciplines

- We developed new efficient algorithms for solving stochastic games, which have applications in other fields such as economics and biology.

- We contributed to scientific interdisciplinary information sharing through collaboration and major contribution to the framework of the Kepler Scientific Workflow project.

- We have shown that hybrid systems theory can be coupled with Lagrangian methods to produce reduced state-space expressions of computationally difficult problems, such as the motion of a bipedal walker.

5.3. Human Resource Development

Several panels in important conferences and workshops pertinent to embedded systems (e.g., DAC, ICCAD, HSCC, EMSOFT, CASES, and RTSS) have pointed out the necessity of upgrading the talents of the engineering community to cope with the challenges posed by the next generation embedded system technology. Our research program has touched many graduate students in our institutions and several visiting researchers from industry and other Universities so that they now have a deep understanding of embedded system software issues and techniques to address them.

Specifically, our directors played a major role in the development of workshops and briefings to executives and researchers in the avionics industry to motivate increased research spending due to an anticipated drop in research funds available to train graduates in embedded software and embedded systems. One particular intersection with our efforts is the Software Producibility Initiative out of the Office of the Secretary of Defense.

The industrial affiliates to our research program are increasing and we hope to be able to export in their environments a modern view of system design. Preliminary feedback from our partners has underlined the importance of this process to develop the professional talent pool.

5.4. Integration of Research and Education

In this report, we have touched multiple times on research and education especially in the outreach section. In addition, there has been a strong activity in the continued update of the undergraduate course taught at Berkeley on the foundations of embedded system design. The graduate program at Berkeley and at Vanderbilt has greatly benefited from the research work in the ITR. EE249 at Berkeley has incorporated the most important results thus far obtained in the research program. EE 290 A and C, advanced courses for PhD students, have featured hybrid system and the interface theories developed under this project. EE219C, a course on formal verification, has used results from the hybrid theory verification work in the program. Finally, many final projects in these graduate courses have resulted in papers and reports listed in this document. The course EE291E on Hybrid Systems: Computation and Control is jointly taught at Berkeley and Vanderbilt and is benefiting a great deal from comments of students as far as the development of new text book material.

In addition to the influence on graduate students, we have endeavored to show hybrid and embedded systems as emerging research opportunities to undergraduates. We have also demonstrated that for advanced undergraduates these topics are not out of place as senior design courses, or advanced topics courses, which may in the future lead to the integration of these as disciplines in engineering across a broader reach of universities.

Due to the benefit of a large center, we were able to use the model CHESS espoused in developing the summer program for SUPERB in influencing how the remainder of the program would be run this year, paying special attention to defining an undergraduate research project which could then be matured by a graduate student. This year the SUPERB program produced a two-semester undergraduate course which attracted electrical and mechanical engineering undergraduates to use hybrid systems theory for application in bipedal walking. In addition, there were several conference papers written which are in publication at this time; these included the undergraduates as authors.

5.5. Beyond Science and Engineering

Embedded systems are part of our everyday life and will be much more so in the future. In particular, wireless sensor networks will provide a framework for much better environmental monitoring, energy conservation programs, defense and health care. Already in the application chapter, we can see the impact of our work on these themes. In the domain of transportation systems, our research is improving safety in cars, and foundationally improving control of energy conserving aspects such as hydrocarbon emissions. Future applications of hybrid system technology will involve biological systems to a much larger extent showing that our approach can be exported to other field of knowledge ranging from economics to biology and medicine. At Berkeley, the Center for Information Technology Research in the Interest of Society is demonstrating the potential of our research in fields that touch all aspects of our life. Some key societal grand challenge problems where our ITR research is making a difference includes health care delivery, high confidence medical devices and systems, avionics, cybersecurity, and transportation.