### 1 Status of the Effort

We have reached the following major milestones toward the compositional design of high confidence embedded control systems on computational and communication platforms.

1. We have developed a modular code generation mechanism for hierarchical SDF models. As hierarchical SDF models are themselves not compositional, we proposed so-called DSSF profiles, which form a compositional abstraction of composite actors that can be used for modular compilation. Our method guarantees maximal reusability, but it also allows to tradeoff reusability for greater modularity.
2. Including semantic information in models helps to expose modeling errors early in the design process. We have developed a correct, scalable and automated method to infer semantic properties using lattice-based ontologies, given relatively few manual annotations. We have demonstrated the approach on a nontrivial Ptolemy II model of an adaptive cruise control system. The paper presenting this work [2] was the recipient of the MODELS 2009 Distinguished Paper Award.
   1. We have released Ptolemy II 8.0.beta on February 26, 2010 including the following new or greatly improved features:
      * Model Transformation -a framework for the analysis and transformation of actor models using model transformation techniques
      * Ptera (Ptolemy Event Relationship Actor) Domain
      * Causality Analysis: Updates to our non-conservative causality analysis for modal models within discrete-event (DE) systems
      * Continuous and Modal Domains: a substantial rework of modal models and the underlying finite state machine infrastructure to make them work predictably and consistently across domains.
3. We have enriched a significant subset of hierarchical Ptolemy II DE models with formal verification capabilities using Real-Time Maude as back-end. To this end, we formalized in Real-Time Maude the semantics of a subset of hierarchical Ptolemy II DE models, and used the code generation infrastructure of Ptolemy II to automatically synthesize a Real-Time Maude verification model from a Ptolemy II design model. This enables a model-engineering process that combines the convenience of Ptolemy II DE modeling and simulation with formal verification in Real-Time Maude.

### 2 Accomplishments and New Findings

#### 2.1 Hybrid and Embedded Systems Theory

Finite State Machines and Modal Models in Ptolemy II Finite-state machines (FSMs) and modal models provide a very expressive way to build up complex model behaviors. As a consequence of this expressiveness, it takes some practice to learn to use them well. Edward Lee has compiled a technical memorandum [1] that describes the usage and semantics of FSMs and modal models in Ptolemy II.

FSMs are actors whose behavior is described using a finite set of states and transitions between the states. The transitions between the states are enabled by guards, which are boolean-valued expressions that can reference inputs to the actor and parameters in scope. The transitions can produce outputs and can update the value of parameters in scope. Modal models extend FSMs by allowing states to have refinements, which are hierarchical Ptolemy II models. The refinements may themselves be FSMs, modal models, or any composite actor containing a director compatible with the domain in which the modal model is being used. On the basis of several examples, the memorandum describes the operational semantics, the practical usage, and the semantics of time in modal models.

Each of the figures in the document corresponds to an executable Ptolemy II model. To execute and experiment with these models the reader can simply click on the corresponding figures in the document. There is no need to pre-install Ptolemy II or any other software.

#### 2.2 Model-Based Software Design and Verification

##### 2.2.1 Autocoding Embedded Software for Safety Critical Systems (Lee)

*Modular Code Generation for SDF*

Modeling languages with built-in concepts of concurrency, time, I/O interaction, and so on, are particularly suitable in the domain of embedded systems. Languages such as Simulink, UML or SystemC, and corresponding tools, are particularly popular in this domain, for various applications. The tools provide modeling and simulation, but often also code generation and static analysis or verification capabilities, which are increasingly important in an industrial setting. A widespread model of computation in this domain is Synchronous Data Flow (SDF).

To manage complexity, as with other models, SDF models are built in the Ptolemy framework in a modular, hierarchical manner. But hierarchical SDF models are not compositional: a composite SDF actor cannot be represented as an atomic SDF actor without loss of information that can lead to deadlocks. Motivated by the need for incremental and modular code generation from hierarchical SDF models, we have studied the problem of developing a compositional SDF representation.

For this purpose, we propose DSSF profiles. DSSF (Deterministic SDF with Shared FIFOs) forms a compositional abstraction of composite actors that can be used for modular compilation.

1. We have developed algorithms for automatic synthesis of non-monolithic DSSF profiles of composite actors given DSSF profiles of their sub-actors.
2. We show how different tradeoffs can be explored when synthesizing such profiles, in terms of modularity (keeping the size of the generated DSSF profile small) versus reusability (preserving information necessary to avoid deadlocks) as well as algorithmic complexity.
3. We show that our method guarantees maximal reusability and report on a prototype implementation.

*Ptolemy Hierarchical Orthogonal Multi-Attribute Solver*

The Ptolemy Hierarchical Orthogonal Multi-Attribute Solver (PtHOMAS) project (in conjunction with Bosch Research Center, Palo Alto) is focused on enhancing model-based design techniques with the ability to include semantic information about data (what the data means) within models, to check consistency in the usage of data across models, and to optimize models based on inferences made about the meaning of the data.

Including semantic information in models helps to expose modeling errors early in the design process, engage a designer in a deeper understanding of the model, and standardize concepts and terminology across a development team. It is impractical, however, for model builders to manually annotate every modeling element with semantic properties. We have developed a correct, scalable and automated method to infer semantic properties using lattice-based ontologies, given relatively few manual annotations. Semantic concepts and their relationships are formalized as a lattice, and relationships within and between components are expressed as a set of constraints and acceptance criteria relative to the lattice. Our inference engine automatically infers properties wherever they are not explicitly specified. Our implementation leverages the infrastructure in the Ptolemy II type system to get efficient and scalable inference and consistency checking. We demonstrate the approach on a non-trivial Ptolemy II model of an adaptive cruise control system.

The paper “Scalable semantic annotation using lattice-based ontologies” [2] presenting this work was accepted at the MODELS conference, which had an acceptance rate of 15%. It was the recipient of the MODELS 2009 Distinguished Paper Award.

In a joint effort with Bosch, we started refactoring our ontologies implementation. We are taking what we learned about ontologies, doing a ground-up redesign and building a system that will be easier to use, extend and test.

*Verifying Hierarchical Ptolemy II Discrete-Event Models using Real-Time Maude*

Model-based design principles put the construction of models at the center of embedded system design processes. Useful models are executable, providing simulations of system functionality, performance, power consumption, or other properties. Ptolemy II is a well-established modeling and simulation tool, developed at UC Berkeley, that provides a powerful and intuitive graphical modeling language to allow a user to build hierarchical models that combine different models of computations including discrete-event (DE) models, which are explicit about timing behavior of systems. Discrete-event modeling is a time honored and widely used approach for system simulation.

Kyungmin Bae (University of Illinois at Urbana-Champaign), Peter Csaba Olveczky (University of Oslo), Thomas Huining Feng and Stavros Tripakis have enriched a significant subset of hierarchical Ptolemy II DE models with formal verification capabilities using Real-Time Maude as back-end. Real-Time Maude is a high-performance tool that extends the rewriting-logic-based Maude system to support the formal specification and analysis of object-based real-time systems. We formalized in Real-Time Maude the semantics of a subset of hierarchical Ptolemy II DE models, and used the code generation infrastructure of Ptolemy II to automatically synthesize a Real-Time Maude verification model from a Ptolemy II design model. This enables a model-engineering process that combines the convenience of Ptolemy II DE modeling and simulation with formal verification in Real-Time Maude. In particular, Ptolemy users can specify temporal logic properties to be verified without understanding how Ptolemy models are represented in Real-Time Maude. Real-Time Maude allows to verify live-ness properties, that cannot be checked using Ptolemy simulations. Furthermore, the synthesized verification model can be formally analyzed w.r.t. other properties (e.g., determinism, etc.).

#### 2.3 Testing and Experimental Validation

### 3 Personnel Supported

* Professor Edward A. Lee (Faculty, funded elsewhere)
* Man-Kit (Jackie) Leung (Research staff, funded by this contract)
* Christopher Brooks (Software engineer, partially funded)
* Bert Rodiers (Graduate student, funded elsewhere)
* Jan Reineke (Postdoctoral scholar, partially funded)
* Ben Lickly (Graduate student, funded elsewhere)
* Thomas Huining Feng (Graduate student, funded elsewhere)
* Stavros Tripakis (Research scientist, funded elsewhere)
* Dai Bui (Graduate student, funded elsewhere)

### 4 Publications

[1] E. A. Lee, “Finite state machines and modal models in ptolemy ii,” EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-151, Nov 2009. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-151.html

[2] J. M.-K. Leung, T. Mandl, E. A. Lee, E. Latronico, C. Shelton, S. Tripakis, and B. Lickly, “Scalable semantic annotation using lattice-based ontologies,” in *12th International Conference on Model Driven Engineering Languages and Systems*. ACM/IEEE, October 2009, pp. 393–407, (recipient of the MODELS 2009 Distinguished Paper Award). [Online]. Available: http://chess.eecs.berkeley.edu/pubs/611.html

[3] S. Tripakis, D. N. Bui, M. Geilen, B. Rodiers, and E. A. Lee, “Compositionality in synchronous data ﬂow: Modular code generation from hierarchical SDF graphs,” UC Berkeley, Tech. Rep. UCB/EECS-2010-52, May 2010. [Online]. Available: http://chess.eecs.berkeley.edu/pubs/668.html

[4] S. Tripakis, D. N. Bui, B. Rodiers, and E. A. Lee, “Compositionality in synchronous data ﬂow: Modular code generation from hierarchical SDF graphs (poster abstract),” in *ACM/IEEE First International Conference on Cyber-Physical Systems*, R. Rajkumar, Ed., April 2010. [Online]. Available: http://chess.eecs.berkeley.edu/pubs/662.html

[5] K. Bae, P. C. Olveczky, T. H. Feng, and S. Tripakis, “Verifying Ptolemy II discrete-event models using Real-Time Maude,” in *ICFEM ’09: Proceedings of the 11th International Conference on Formal Engineering Methods*, December 2009, pp. 717–736. [Online]. Available: http://chess.eecs.berkeley.edu/pubs/652.html

[6] K. Bae, P. Olveczky, T. H. Feng, E. A. Lee, and S. Tripakis, “Verifying hierarchical Ptolemy II discrete-event models using Real-Time Maude,” UC Berkeley, Tech. Rep. UCB/EECS-2010-50, May 2010. [Online]. Available: http://chess.eecs.berkeley.edu/pubs/667.html

### 5 Interactions/Transitions

#### 5.1 Participation/presentations at meetings, conferences, seminars

* Ben Lickly presented “Scalable Semantic Annotation using Lattice-based Ontologies” at the 12th International Conference on Model Driven Engineering Languages and Systems (MoDELS) in Denver, Colorado, October 2009
* Dai Bui presented a poster about “Compositionality in synchronous data ﬂow: Modular code generation from hierarchical SDF graphs” at the International Conference on Cyber-Physical Systems (ICCPS) in Stockholm, Sweden, April 2010

#### 5.2 Consultative and advisory functions

1. Edward A. Lee

(a) Air Force Research Laboratory, AFRL/RIEA, Rome, NY Brian Romano USAF AFMC AFRL/RIEF

Brian.Romano@rl.af.mil 315-330-4218

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

(b)

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

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

(c)

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

The objective of the SCOS research project is to provide scalable techniques for the composition of subsystems in a system-of-systems (SoS) framework for large, complex applications such as FCS. SCOS has synergy with this MURI project in that it deals with large systems. In particular:

* we are using the EmbeddedCActor to wrap legacy C code
* we are collaborating on work on the Kepler Project
* we are using Graph Transformations on models

#### 5.3 Technology Assists, Transitions, and Transfers

* 8*th* Biennial Ptolemy Miniconference, Thursday, April 16, 2009 in Berkeley, California
  + Ptolemy II 8.0.beta was released on February 26, 2010 including
    - Model Transformation -a framework for the analysis and transformation of actor models using model transformation techniques
    - Ptera (Ptolemy Event Relationship Actor) Domain
    - Causality Analysis: Updates to our non-conservative causality analysis for modal models within discrete-event (DE) systems
    - Continuous and Modal Domains: a substantial rework of modal models and the underlying finite state machine infrastructure to make them work predictably and consistently across domains.

#### 5.4 Honors and Awards

1. Edward Lee

* MODELS Distinguished Paper Award for “Scalable Semantic Annotation using Lattice-based Ontologies” [2]
  + Keynotes:
    - “Model-Based Design for Signal Processing Systems”, IEEE Workshop on Signal Processing Systems (SiPS), October 7-9, 2009, Tampere, Finland.
    - “Beyond Embedded Systems: Integrating Computation, Networking, and Physical Dynamics”, ACM SIGPLAN/SIGBED 2009 Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES), June 19-20, 2009, Dublin, Ireland.