Introduction

Edited and presented by
Edward A. Lee, Professor,
Chair of the EE Division, and
Associate Chair of EECS

Chess Review
May 11, 2005
Berkeley, CA

Chess

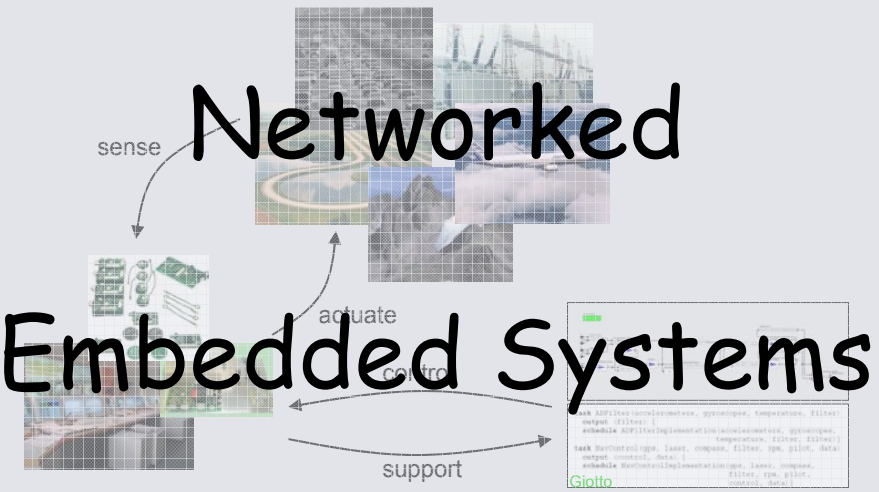
Background on Chess

- Founded in 2002
- Large ITR funded in 2002 with Partners:
  - Vanderbilt (ISIS)
  - University of Memphis
- November, 2004
  - Passed first official NSF review
- Today
  - Informal review, mainly for industrial partners
  - Focus on network embedded systems

Participants

- Faculty:
  - Berkeley: Ruzena Bajcsy, Ras Bodik, Bella Bollobas, Gautam Biswas,Tom Henzinger, Edward Lee, George Necula, Alberto Sangiovanni Vincentelli, Shankar Sastry
  - Vanderbilt: Gautam Biswas, Kenneth Frampton, Gabor Karsai, Kurt Keutzer, John Koo, Janos Sztipanovits, Pravin Varaiya
  - Memphis: Bela Bollobas
- Industrial partners:
  - Agilent, Daimler-Chrysler, General Motors, Hewlett-Packard, Honeywell, Infineon, Toyota

Computation Integrated with the Physical World



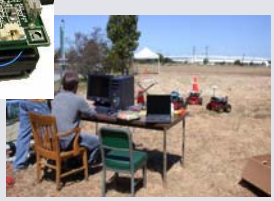
Foundational Research

- The science of computation has systematically abstracted away the physical world. The science of physical systems has systematically ignored computational limitations.
  Embedded software systems, however, engage the physical world in a computational manner.
- We believe that it is time to construct an Integrated Systems Science (ISS) that is simultaneously computational and physical.
  Time, concurrency, robustness, continuums, and resource management must be remarried to computation.
- Mathematical foundation:
  Hybrid Systems Theory, Integrated Systems Science.

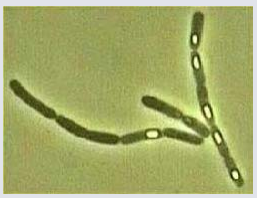
Some Applications Addressed

Automotive

Avionics: UAVs



Systems Biology

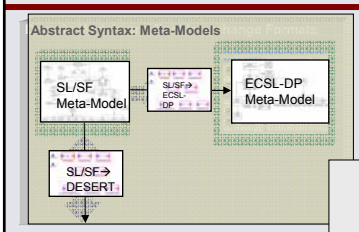


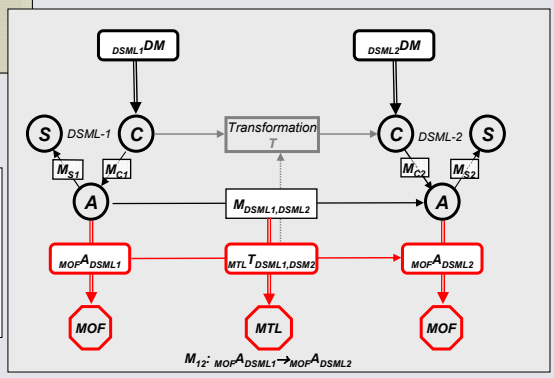
Networked
Embedded Systems

Some of Our Contributions

- Controller synthesis
- Abstract semantics and semantic anchoring
- Model transformation
- Hybrid systems semantics (denotational & operational)
- Refinement verification
- Composition of domain-specific modeling languages
- Schedule carrying code
- Discounted systems theory
- Theories of phase transitions
- Meta frameworks (GME, Ptolemy II, Metropolis)
- Interface checking
- Lazy abstraction
- Education
- Outreach (SUPERB-IT, SiPHER, Escher)

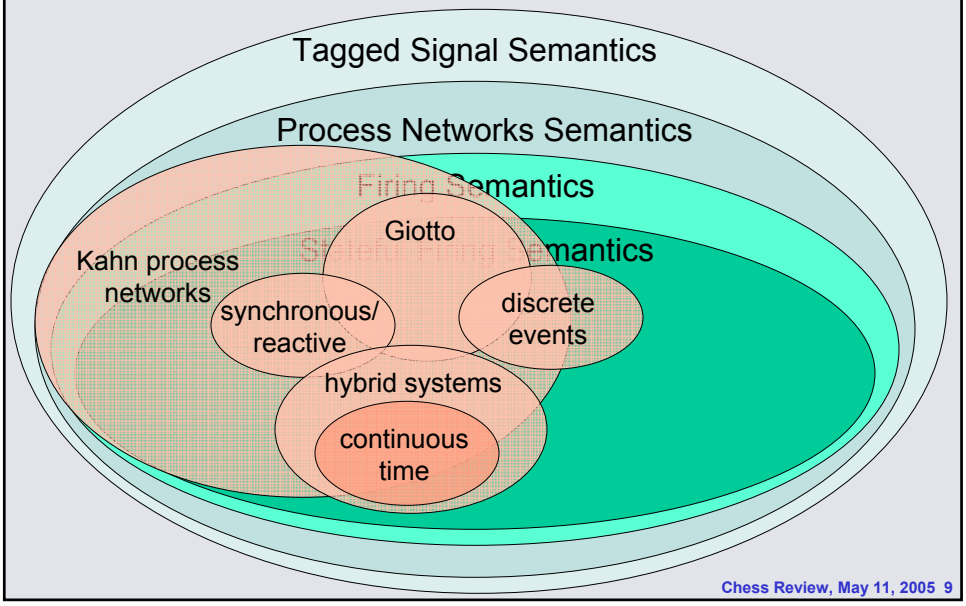
Abstract Syntax Metamodels Driving Model Transformations





- Gives structural semantics for the models
- Set-valued Semantic Domain for the metamodels and transformations

Abstract Semantics



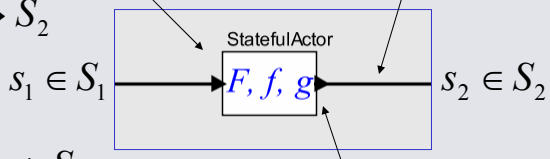
Example of an Abstract Semantics

Stateful Firing Abstract Semantics:

a process still a function from input signals to output signals, but that function now is defined in terms of two functions.

signals are monoids (can be incrementally constructed) (e.g. streams, discrete-event signals).

$F : S_1 \rightarrow S_2$



$f : S_1 \times \Sigma \rightarrow S_2$

$g : S_1 \times \Sigma \rightarrow \Sigma$

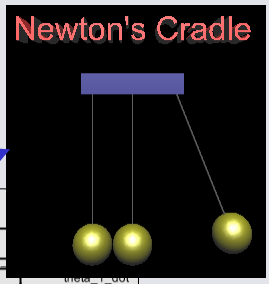
state space

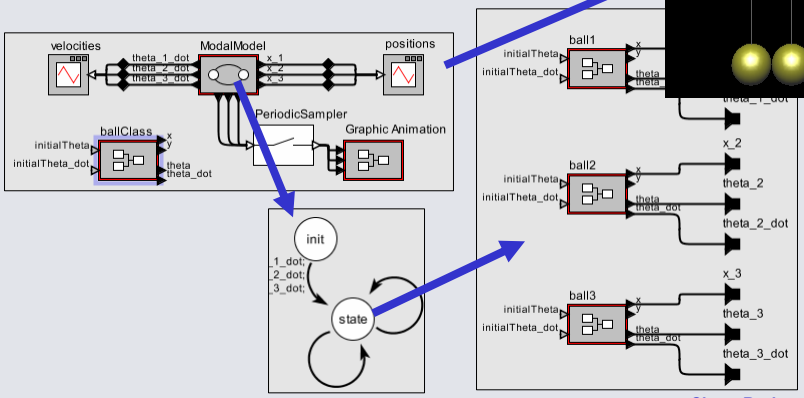
port is still either an input or an output.

The function f gives outputs in terms of inputs and the current state. The function g updates the state.

Leveraging this Abstract Semantics: HyVisual: Computation of Hybrid Systems

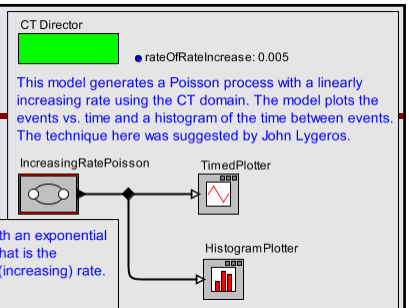
Clean, clear, and rigorous semantics for discrete-event, continuous-time, and hybrid systems [Cataldo, Liu, Matsikoudis, Zheng]

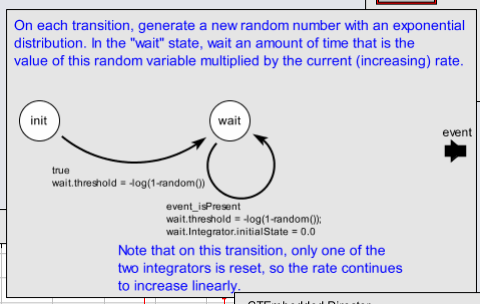




More Leveraging: Stochastic Hybrid Systems

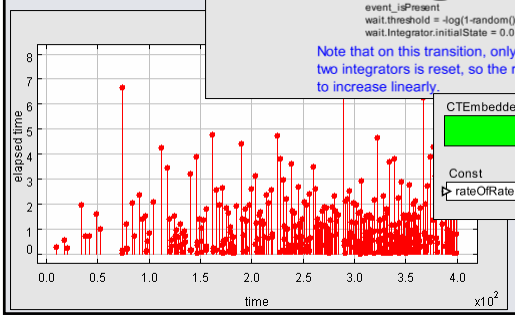
Stochastic hybrid systems in Ptolemy II are Monte-Carlo models of nondeterminism

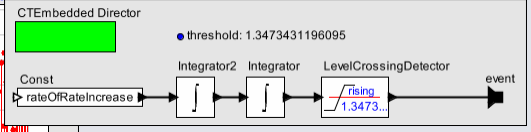




event

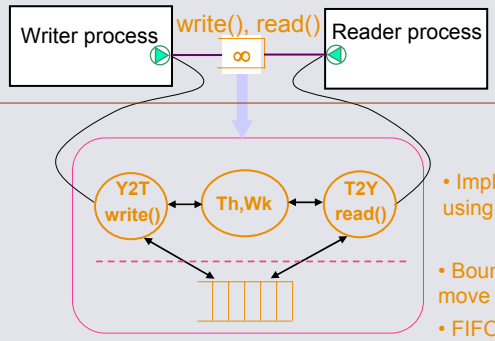
Example of random "spontaneous transitions" by Lee and Zheng, based on suggestion by John Lygeros.





Leveraging a different Abstract Semantics for Refinement Verification in Metropolis

Example: a unbounded FIFO v.s. a bounded FIFO with the finer service.



Unbounded FIFO Level

Bounded FIFO Level

- Implement the upper level services using the current services
- Bounded FIFO API, e.g. release space, move data
- FIFO width and length parameterized

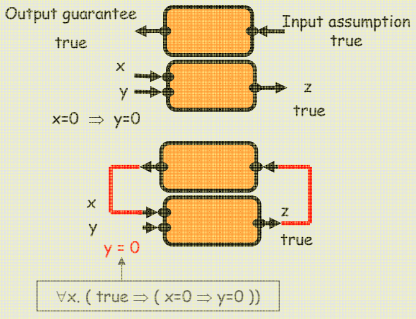
: **refinement relation**

- Metropolis represent both levels of abstraction explicitly, rather than replacing the upper level.
- Refinement relation is associated with properties to preserve through the refinement.

Thanks to Doug Densmore

Chic: A Tool for Checking Interface Compatibility

Interface: Expresses assumptions made by module about environment, and guarantees made by module if assumptions are satisfied.
Interface = Behavioral type



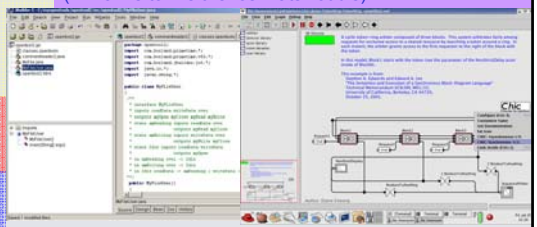
Compatibility checking is a game between System and Environment; winning strategy of Environment gives correct way to use System.

Software Module interfaces allow pushdown analysis to check safety properties of recursive software components.

Resource interfaces: automata-based type system for compositional resource-aware analysis of embedded software. eg. Node Limit Interfaces express requirements like mutex, limited buffer size, limited peak power. Path Limit Interfaces express requirements like limited battery capacity. Compositional and scalable.

Web Service interfaces allow checking temporal properties of interaction between service components.

Chic 1.1 is available as a plug-in for JBuilder, Ptolemy*. Implemented in Java. Supports static, dynamic (including pushdown) and resource interfaces. Support for web service interfaces is under development.
(* Thanks to Eleftherios Matsikoudis)



Download Chic 1.1 today !! http://www.eecs.berkeley.edu/~tah/Chic/

Thrust 1 Hybrid Systems

- Deep Compositionality
  - Assume Guarantee Reasoning for Hybrid Systems
  - Practical Hybrid System Modeling Language
  - Interface Theory for hybrid components
- Robust Hybrid Systems
  - Bundle Properties for hybrid systems
  - Topologies for hybrid systems
  - Stochastic hybrid systems
- Computational hybrid systems
  - Approximation techniques for H-J equations
  - Synthesis of safe and live controllers for hybrid systems
- Phase Transitions

Thrust II: Model Based Design

- Composition of Domain Specific Modeling Languages
  - Meta Modeling
  - Components to manipulate meta-models
  - Integration of meta-modeling with hybrid systems
- Model Synthesis Using Design Patterns
  - Pattern Based Modal Synthesis
  - Models of Computation
  - Design Constraints and Patterns for MMOC
- Model Transformation
  - Meta Generators
  - Scalable Models
  - Construction of Embeddable Generators

Thrust III:Advanced Tool Architectures

- Syntax and Synthesis
  - Semantic Composition
  - Visual Concrete Syntaxes
  - Modal Models
- Interface Theories
- Virtual Machine Architectures
- Components for Embedded Systems

Thrust IV: Applications

- Embedded Control Systems
  - Avionics
  - Veitronics
  - Wireless Embedded Systems
- Embedded Systems for National/Homeland Security
  - Air Traffic Control
  - UAVs/UGVs
- Networks of Distributed Sensors
- Stochastic Hybrid Systems in Systems Biology
- Hybrid Models in Structural Engineering
  - Active Noise Control
  - Vibration damping of complex structures

Thrust V: Education and Outreach

- Curriculum Development for MSS
  - Lower Division
  - Upper Division
  - Graduate Courses
- Undergrad Course Insertion and Transfer
  - Goals and ABET requirement
  - New courses for partner institutions (workshop held March 1st 2003, Summer 2004)
  - Introduction of new courses (will be replacing control course at upper division level by embedded control course jt with San Jose State)
  - CHESS-SUPERB/ Summer Program in Embedded Software Research SIPHER program (6 + 4 students in Summer 03, 3 + 5 in Summer 04)
- Graduate Courses
  - EECS 249 Design of Embedded Systems: Models, Validation, and Synthesis
  - EECS 290N Concurrent Models of Computation for Embedded Software
  - EECS 291E/ME 290S Hybrid Systems: Computation and Control

Outreach Continued

- Interaction with EU-IST programs
  - Columbus (with Cambridge, l'Aquila, Rome, Patras, INRIA)
  - Hybridge, Hycon (with Cambridge, Patras, NLR, Eurocontrol, Brescia, KTH)
  - ARTISTE, ARTIST-2: Educational Initiatives (Grenoble, INRIA, ETH-Zurich)
  - RUNES, new EU-IST program in network embedded systems (Ericsson, KTH, Aachen, Brescia, Pisa, Patras, …)
- Foundation of non-profit ESCHER
  - Interaction with F-22/JSF design review teams
  - Secure Networked Embedded Systems: SCADA systems

The Big Question: How to Give Semantic Meta Models that are Usefully Manipulable

Key ideas guiding us:

- Abstract semantics
- Ptolemy II directors
- Metropolis quantity managers
- The Metropolis language of constraints
- Interface theories
- Behavioral type systems
- Temporal logics (e.g. TLA)
- Set-valued semantics
- …

Related New Initiative: TRUST STC

Lead: Shankar Sastry

TRUST:Team for Research in Ubiquitous Secure Technologies

- Security Science (**Schneider**)
  - Software Security (**Mitchell**)
  - Trusted Platforms (**Boneh**)
  - Applied Cryptography Protocols (**Wagner**)
  - Network Security (**Joseph**)
- Systems Science (**Schmidt**)
  - Interdependency Modeling and Analysis (**Anantharam**)
  - Secure Network Embedded Systems (**Wicker**)
  - Model Based Integration of Trusted Components (**Sztipanovits**)
  - Secure Information Management Tools (**Birman**)
- Social, Economic and Legal Considerations (**Samuelson**)
  - Economics, Public Policy and Societal Challenges (**Varian**)
  - Digital Forensics and Privacy (**Tygar**)
  - Human computer Interfaces and Security (**Reiter**)

Starts June 1, 2005. $19M over 5 years.

Related New Initiative: Adaptive Networked Infrastructure – Proposed ERC

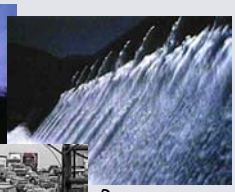
Core partners: Berkeley (lead), Cornell, Vanderbilt

Outreach partners: San Jose State, Smith, Tennessee Tech, UC Davis, UC Merced.

Lead: Edward A. Lee

Enabling technologies: wireless networked embedded systems with sensors and actuators

Approach: Engineering methods for integrating computer-controlled, networked sensors and actuators in societal-scale infrastructure systems.



Resource management test beds:
- electric power
- transportation
- water

The ANI ERC

Deliverables: Engineering Methods, Models, and Toolkits for:
- design and analysis of systems with embedded computing
- computation integrated with the physical world
- analysis of control dynamics with software and network behavior
- programming the ensemble, not the computer
- computer-integrated systems oriented engineering curricula