HETEROGENEOUS CONCURRENT MODELING AND DESIGN IN JAVA

Patricia Derler, Edward Lee
University of California, Berkeley

Invited Talk
International Symposium on System-on-Chip 2011
Tampere, Finland, November 2
MODELING FORMALISMS AND TOOLS

UML [Wikipedia Commons]
AADL [AADL community Wiki]
Labview [http://www.ni.com/labview/]
Matlab/Simulink [http://www.mathworks.com]

Fault Trees [Wikipedia]
IEC 61499

Ptolemy II
WHAT DOES THIS MODEL DESCRIBE?

When do components execute?

How many tokens can I/do I have to send?

What kind of interaction?

What kind of data is transmitted?

What about time? (execution time, timing requirements)

How many tokens can I/do I have to consume?

We need a clear semantics!
FOCUS IN THE PTOLEMY PROJECT

- Modeling, simulation, and design of concurrent, real-time, embedded systems
- Executable models rather than descriptive models
- Semantics matters (more than syntax)
- Useful semantics imply constraints on designers
- Heterogeneity may be better than generality

Freedom from choice!
PTOLEMY II
Our Laboratory for Actor-Oriented Models

Director from an extensible library defines component interaction semantics or “model of computation”

Type system for transported data

Visual editor supporting an abstract syntax

Extensible, behaviorally-polymorphic component library
MODEL OF COMPUTATION (MOC)

A MoC describes:

- What constitutes an actor
- Concurrency mechanism
- Communication mechanism
MODEL OF COMPUTATION

- Process Network (PN)
- Rendezvous
- Synchronous Dataflow (SDF)
- Finite State Machine (FSM)
- Discrete Event (DE)
- Continuous
- Push/pull
- Distributed DE
- Dynamic Dataflow
- Discrete Time
- Distributed PN
- Timed Multitasking
- Giotto
- 2-D and 3-D graphics
- ...

Patricia Derler -2011
• Implementation: Each actor executes in its own Java thread, so on multi-core machines they can execute in parallel.

• Explicit support for non-determinism


PROCESS NETWORK

Author: Edward A. Lee

SoC 2001 - Invited Talk - Patricia Derler
The process that first reaches a rendezvous point will stall until the other process also reaches the same rendezvous point.

- Like PN, this domain supports explicit nondeterminism.

BARRIER SYNCHRONIZATION WITH RENDEZVOUS

Author: Edward A. Lee
DATAFLOW

restricted special cases of PN


Produce a fixed number of tokens on the outputs

Consume a fixed number of tokens from every input


SYNCHRONOUS DATAFLOW

• Strong, decidable, formal properties

• Deadlock and boundedness can be statically analyzed

• Execution order of components (schedule, even for parallel execution) is statically scheduled.

• Potential for very high quality code generation

• Timed or untimed variants

• The dynamic dataflow (DDF) domain is more flexible, but computes schedules on the fly
A multirate SDF model. The Spectrum actor requires 256 tokens to fire, so one iteration of this model results in 256 firings of Sinewave, Channel, and SequencePlotter, and one firing of Spectrum.
DATAFLOW VARIANTS

- Computation graphs [Karp and Miller, 1966]
- Static dataflow [Dennis, 1974]
- Dynamic dataflow [Arvind, 1981]
- Structured dataflow [Matwin & Pietrzykowski, 1985]
- K-bounded loops [Culler, 1986]
- Synchronous dataflow [Lee & Messerschmitt, 1986]
- Structured dataflow and LabVIEW [Kodosky, 1986]
- PGM: Processing Graph Method [Kaplan, 1987]
- Synchronous languages [Lustre, Signal, 1980’s]
- Well-behaved dataflow [Gao, 1992]
- Boolean dataflow [Buck and Lee, 1993]
- Multidimensional SDF [Lee, 1993]
- Cyclo-static dataflow [Lauwereins, 1994]
- Integer dataflow [Buck, 1994]
- Bounded dynamic dataflow [Lee and Parks, 1995]
- Parameterized Dataflow [Bhattacharya et al. 00]
- Structured Dataflow [Kodosky 86, Thies et al. 02]
- Teleport Messages [Thies et al. 05]
DISCRETE EVENT

Send timestamped events

Consume events with timestamp equal to current model time

Maintain global event queue


Actors are fired sequentially
• What if they do not return from the firing?
• => ThreadedComposite

DISCRETE EVENT

• Actors process events in chronological order

• The output events produced by an actor are required to be no earlier in time than the input events that were consumed. In other words, actors in DE are causal.

• During each iteration
  • events with the smallest time stamp are removed from the global event queue
  • and then the destination actor is fired.
Messages have a (semantic) time, and actors react to messages chronologically. Merge now becomes deterministic.
WHAT ABOUT SIMULTANEOUS EVENTS?

• Superdense time model

  Timestamp \((t, n)\)
  
  \(t\): model time, time at which event occurs
  \(n\): microstep (also index), sequencing of events that occur at the same time

  If an event with timestamp \((t, n_1)\) causes an event with timestamp \((t, n_2)\) then \(n_1 < n_2\)

• Sequences of zero-time events in discrete signals


WHAT ABOUT ORDER OF EVENTS?

• Order of firings has to be strictly defined (topological sort)

• Level for each actor = length of the longest upstream path of actors that may produce events on which it depends that have the same time stamp.

• Feedback: delay actors: level 0, can only consume events in postfire
WIRELESS

• Derived from DE

• Wireless actors communicate via channels

• Allows for modeling of
  • Power loss
  • Interference
  • Noise
  • Occlusion
CONTINUOUS

\[ x(t) = x_0 + \int_{t_0}^{t} \dot{x}(\tau) d\tau \]

- Initial state of the integrator
- Start time of the director
- Input signal to the integrator

Special actors that represent integrators are connected in feedback loops in order to represent the ODEs.

Also allowed: actors that produce discrete events.

Receiver: buffer with size one. It contains the value of the continuous function of the corresponding connection at a specific time instant.

Uses the superdense time model for discontinuities in continuous-time signals.


Liu, J.: Continuous time and mixed-signal simulation in Ptolemy II. Memo M98/74, UCB/ERL, EECS UC Berkeley, CA 94720 (July 1998)
CONTINUOUS

Execution of a model: computation of a numerical solution to the ODEs.

• In an iteration, time is advanced (amount determined by solver)
• At each instant, the director computes a fixed point for all signal values

• Director speculatively executes actors

• If the time step is sufficiently small (key events such as level crossings, mode changes, or requested firing times are not skipped over), then the director commits the time increment and continues

The Continuous interoperates with all other timed Ptolemy II domains.
FUEL TANK MODEL

- **Continuous Director**
  - **Const**: 1.0

- **Tank**
  - **desiredInFlow**
  - **desiredOutFlow**
  - **availableOutFlow**
  - **actualInFlow**
  - **fullIndicator**
  - **emptyIndicator**

- **ModalModel_1**
  - **capacity**: 10.0
  - **initialLevel**: 2.0
  - **actualLevel**: initialLevel

- **TimedPlotter**

- **ActualInFlow**

- **LevelCrossing**
  - **rising**: 10.0
  - **falling**: 0.0

- **actualInFlow**
- **level**
- **actualOutFlow**
- **desiredOutFlow**

Graph shows the flow of fuel over time with different indicators.
FINITE STATE MACHINE

- Not actor-oriented
- Ports are not used for communication but for sequential control

Entities: states (initial, final)

Relations: transitions between states

- default, guarded
- actions (e.g. set output values)
- preemptive, reset
FINITE STATE MACHINE

• Start of the execution: select initial state

• In the fire() method, the actor
  • 1. reads inputs;
  • 2. evaluates guards on outgoing transitions of the current state;
  • 3. chooses a transition whose guard evaluates to true; and
  • 4. executes the output actions on the chosen transition, if any.

• In the postfire() method, the actor
  • 5. executes the set actions of the chosen transition; and
  • 6. changes the current state to the destination of the chosen transition.
MODAL MODELS

- Extend FSMs by allowing states to have refinements

- States are modes of operation

- Output actions: Mealy

- Outputs in submodels: Moore
THERMOSTAT

non-determinism to model error
HETEROGENEITY

Amorphous

Color is an interactive style (KPN, publish - subscribe, procedure call, ...)

Hierarchical

Color is an MoC, with constraints. The meaning is clear at each level of the hierarchy
HETEROGENEITY

• MoC must be compositional
• Turns itself into a component that in turn may be aggregated with other models, possibly under a different MoC

• Some examples:
  • execute entire iteration
  • use for complex, untimed computation
  • period param to execute periodically
  • rarely makes sense, SDF only fires DE component at multiples of its period (not when DE component requests)
  • ok, if period of SDF = 0; then firing requests from inner DE are delegated
HETEROGENEITY

• MoC must be compositional

• Turns itself into a component that in turn may be aggregated with other models, possibly under a different MoC

• Some examples:

  • does rarely make sense because the notion of an iteration is not defined for PN

  • when FSM goes into a final state the model terminates

  • FSM is reactive, reacts in 0 time
HETEROGENEITY

• MoC must be compositional

• Turns itself into a component that in turn may be aggregated with other models, possibly under a different MoC

• Some examples:

  - interesting behavior when there are time delays in modes
  - time stands still in inactive modes

Cyber-Physical Systems

  - physical view
  - emphasis on physical plant

  - cyber/computation view
  - emphasis on discrete controller
HYBRID SYSTEMS

• Combining Continuous with Discrete mode changes (via FSMs/modal models)


CYBER-PHYSICAL SYSTEMS

Multiple computers, comprising of sensors and actuators, connected on a network that act and react on events to meet timing constraints.
PTIDES

• DE based model of computation

• Model time for model semantics, physical time to explicitly model timing behavior

• Relate physical time to model time at specific points in the model
A PTIDES MODEL

- increase time stamp
- platform time progresses
- time stamp = platform time

Platform 1

Sensor 1

model time delay d4

Computation 1

Platform 2

Sensor 2

model time delay d5

Platform 3

model time delay d1

model time delay d2

Merge

Actuator 1

Physical plant

Physical interface

network fabric

Local Event Source

Computation 4

response time \leq \text{logical time delay} = d4 + d2

and over network

platform time progresses

platform time progresses

platform time progresses

platform time progresses

platform time progresses

platform time progresses

platform time progresses
A CYBER-PHYSICAL SYSTEM MODEL

Platform 1

Platform 2

Platform 3

DE Director

Ptides Director

Sensor 1

model time delay d4

Computation 1

NI 1

Ni 2

Computations

model time delay d1

Ptides Director

Sensor 2

model time delay d5

Computation 2

NI 3

NI 5

Network fabric

Local Event Source

model time delay d3

trigger

Continuous Director

Plant

Actuator 1

merge

Computation 4

SoC 2001 - Invited Talk - Patricia Derler
EXPLICIT SPECIFICATION OF TIME

response time = d4 + d2
EXPLICIT SPECIFICATION OF TIME

• Behavior independent of distribution

response time = d4 + d2
Safe-to-Process Analysis

Event = (timestamp, event)

Can an event at this port arrive with a smaller timestamp later in the execution?
Evaluate other events, Sensor input patterns, delays, dependencies, actor topologies, scheduling strategy, ...
PARALLEL EXECUTION
PARALLEL EXECUTION

Single Core
PARALLEL EXECUTION

Multi Core
SUMMARY

• Semantics of models

• Models of computation
  • Process Network
  • Rendezvous
  • Synchronous Dataflow
  • Discrete Event
  • Continuous

• Heterogeneity

• Modeling Cyber-Physical Systems