Heterogeneous Systems and Multi-Paradigm Modeling

Cécile Hardebolle, Frédéric Boulanger, Idir Ait Sadoune
Who are we?

- Supélec = leading engineering school ("Grande Ecole") in information sciences and energy
  - **Degree courses:** 460 students graduating each year (engineering diploma)
  - **Continuing education**
  - **Research & development:** Supélec Systems Science (E3S)
    (automatic control, signal processing, radio communications, electromagnetism, power systems, computer science)

- **Department of Computer Science =**
  - research & education department
    - **Personalization:** adaptive hypermedia, guided web queries (4 + 4 PhD students)
    - **Optimization of high-performance networks:** (2 + 2 PhD students)
    - **Modeling techniques for heterogeneous systems:** (6 + 4 PhD students)
Questions

① What is heterogeneity?
Heterogeneity at the system level

Physical world

Mechanics, aerodynamics

Electronics

Control systems

Software platforms

Hardware platforms

Energy

Networks

Sensors/actuators
Heterogeneity at the model level

- Combination of components of different natures (signal processing, electronics, control...)
  - Composition of models

- Several abstraction levels
  - Refinement of models

- Orthogonal points of view
  - Models of functional and extra-functional properties/behavior

- Different activities and goals during a project
  - Models for different kind of analysis
Heterogeneity in ModHel'X

- Focus on the heterogeneity of the components of a system:
  - Heterogeneous components ➞ heterogeneous design paradigms
  - Interaction among components + environment ➞ model composition

- The problem we try to address =
  How to compose models that are written using different modeling languages in order to be able to reason globally on a system under design?

- Experimental platform = ModHel'X
The power window example

- Timed FSM (TFSM) model
- Synchronous Dataflow (SDF) model
- Or Continuous Time (CT) model
- Discrete Events (DE) model

Switch

Controller

Window

Bus
Questions

1. What is heterogeneity?
2. How to represent a modeling paradigm in a form that is “composable”?
Model of Computation

- Represents the **semantics** of a modeling language
- Provides the rules for **interpreting** a model

![Diagram showing relationships between FSM and CSP]

- FSM
- CSP
**Model = structure + MoC**

- The **structure** of a model is a set of **interconnected blocks** (black boxes)
- A **MoC** is used to provide an **interpretation (semantics)** of that structure
MoCs currently available in ModHel'X

- **Discrete Events (DE)**
  - Exchange of events \(\langle \text{value, date} \rangle\)
  - \(\approx\) Network messages

- **Synchronous Data Flow (SDF)**
  - Flows of sampled data
  - Multi sample rate
  - \(\approx\) Simulink block diagrams

- **Timed Finite State Machines (TFSM) [+ FSM + *Charts]**
  - Timed transitions: “after(T)”
  - \(\approx\) very simplified UML's Stateflow

- **Petrinets**
The power window example (again)

Controller

Timed FSM (TFSM) model

Switch

Bus

Discrete Events (DE) model

Window

Composition?

Synchronous Dataflow (SDF) model

Or Continuous Time (CT) model
Questions

1. What is heterogeneity?
2. How to represent a modeling paradigm in a form that is “composable”?
3. How to compose models that use different modeling paradigms?
Composition of heterogeneous models

Extract of the model of the window:
up mode with up endstop detection
Composition of heterogeneous models

- “Interface blocks” are used to embed a model into a block
  ➔ Support for heterogeneity through hierarchy

Extract of the model of the window:
up mode with up endstop detection
What is adaptation?

- Adaptation of data
  - Forms
  - Values

- Adaptation of control flow
  - “Moments” at which “things” happen

- Adaptation of time notions
  - Time scales
  - Time forms (seconds, revolutions, centimeters…)

---

16 Heterogeneous Systems and Multi-Paradigm Modeling 25-nov-11
Outline

1. What is heterogeneity?
2. How to represent a modeling paradigm in a form that is “composable”?
3. How to compose models that use different modeling paradigms?
4. What is the benefit of modeling the adaptation explicitly and apart from the models?
The window model in PtolemyII

Model of the window in “open loop”: up mode with up endstop detection
The window model in PtolemyII

- **Default adaptation:**
  - The SDF model reacts only when events are processed in DE
  - DE events are produced in the DE model each time the SDF model reacts

- Changing the adaptation means **modifying one of the two models**
Adapted model in PtolemyII

Works but the original model has been modified…
Questions

① What is heterogeneity?
② How to represent a modeling paradigm in a form that is “composable”?
③ How to compose models that use different modeling paradigms?
④ What is the benefit of modeling the adaptation explicitly and apart from the models?
Key points

- Our approach:
  - Models of Computation (MoCs) for representing the semantics of design paradigms
  - Semantic adaptation for composing heterogeneous models using hierarchy

- Goals of ModHel'X:
  - Extensible set of MoCs
  - Explicit, customizable and modular semantic adaptation between hierarchical models
Current research directions

- **Modeling MoCs**
  - Imperative form ➔ execution
  - Declarative form ➔ verification & validation
  - Variants of a MoC? Reusability of (parts of) a model of a MoC?

- **Modeling Semantic Adaptation**
  - CCSL constraints to describe adaptation of time and control
  - Language to describe adaptation of data
  - Patterns of adaptation

- **Multi-view modeling**

- **Heterogeneous model testing**
MERCI!
THANK YOU!

(FRAPPE)