ModHel'X & Multi-Paradigm Modeling

CAMPaM 2011

Cécile HARDEBOLLE
cecile.hardebolle@supelec.fr

Goal of the talk

- Present our general approach of multi-paradigm modeling
  (= heterogeneous modeling)

- Present the basics of ModHel'X
  - Hierarchical model composition
  - « Glue » between heterogeneous models ← FOCUS
  - Hierarchical model execution

- Illustrate the use of ModHel'X on the power window example

Contents

1. A few words on Supélec
2. Heterogeneous modeling
3. Basics of ModHel'X
4. Power Window example and demo

Supélec

- Leading engineering school ("Grande Ecole") in information sciences and energy

- Threefold mission:
  - Degree courses
    440 students graduating each year (engineering diploma)
  - Continuing education
  - Research & development: Supélec Systems Science
    (automatic control, signal processing, radio communications, electromagnetism, power systems, computer science)

- Three sites in France: Paris, Rennes, Metz

- Funding: 50% public, 50% private

Department of Computer Science

- Education and research department

- Research around three topics:
  - Personalization: adaptive hypermedia, guided web queries
    - 4 researchers + 4 PhD students
  - Optimization of high-performance networks
    - 2 researchers + 2 PhD students
  - Modeling techniques for heterogeneous systems
    - 6 researchers + 4 PhD students

Contents

1. A few words on Supélec
2. Heterogeneous modeling
3. Basics of ModHel'X
4. Power Window example and demo

Context

- Modeling complex systems
  - To validate design choices (+ validate against specification)
    - Simulation of global behavior for testing
    - Model checking…
  - To evaluate non-functional aspects (power, performances...)

- Application fields
  - Embedded systems
    - Example: EDONA project → automotive application (power window)
  - Ambient intelligent systems
    - Example: CBDP project → sensor & actuator modeling

Complex systems are heterogeneous

- Combination of components of different natures (signal processing, electronics, automatic control...)
  - Hierarchical composition of models
- Several abstraction levels
  - Refine system description
- Orthogonal points of view
  - Functional and non-functional properties
- Different activities and goals during a project
  - Different stages in the V-Model development cycle

Main research direction

- Chosen problem =
  How to compose models written using different modeling languages
  in order to be able to reason globally on a complex system
  that one wants to design?

- Chosen approach =
  - Behavioral models (not only architecture)
  - Notion of Model of Computation (MoC)
    - As in PtolemyII [Lee, Berkeley]
  - Hierarchical composition of models & black box approach
  - Simulation

- Prototype tool called ModHel'X

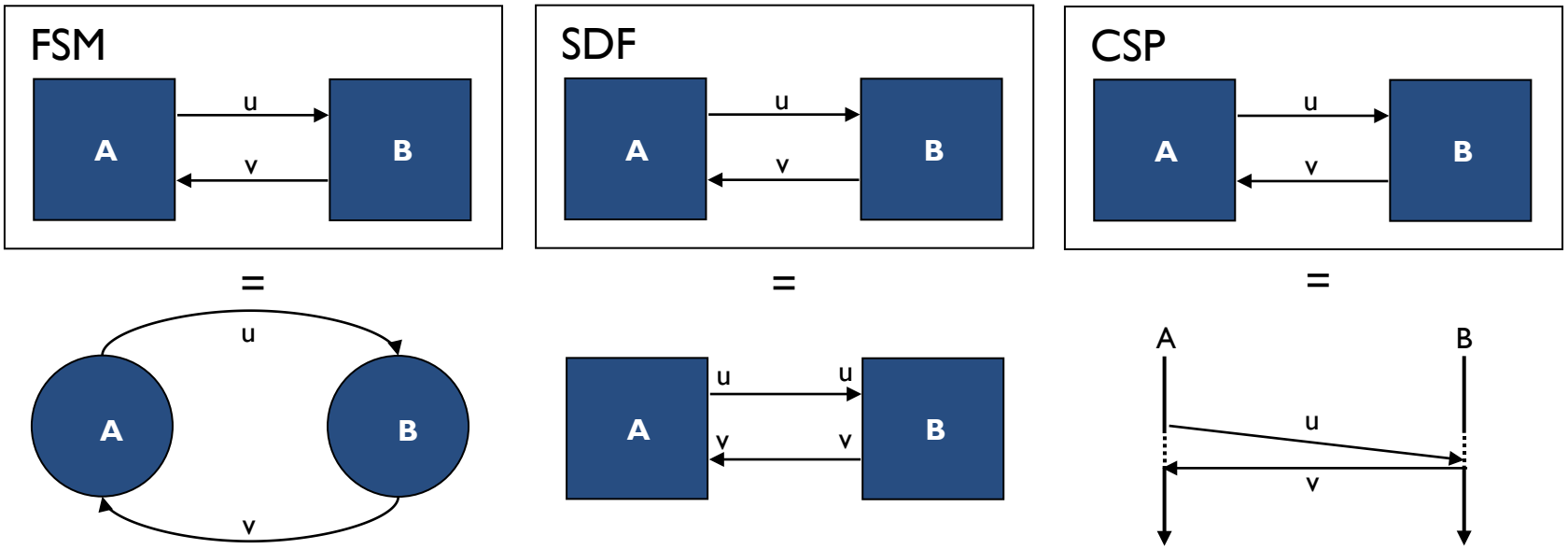
- Application to embedded systems, multi-view modeling, model testing…

Contents

1. A few words on Supélec
2. Heterogeneous modeling
3. Basics of ModHel'X
4. Power Window example and demo

Notion of Model of Computation

- How to interpret a block diagram?



Execution platform for heterogeneous systems

- "Universal" meta-model:
  - Structure: interconnected components (black boxes → block diagram)
  - A MoC applies semantics to the structure

Model = Structure + MoC

- Execution algorithm
  - Succession of observations of the system
  - Generic abstract algorithm specialized for each MoC
    - schedule, update, propagate

- Explicit semantic adaptation between heterogeneous models

ModHel'X platform
Based on Eclipse (plugins)

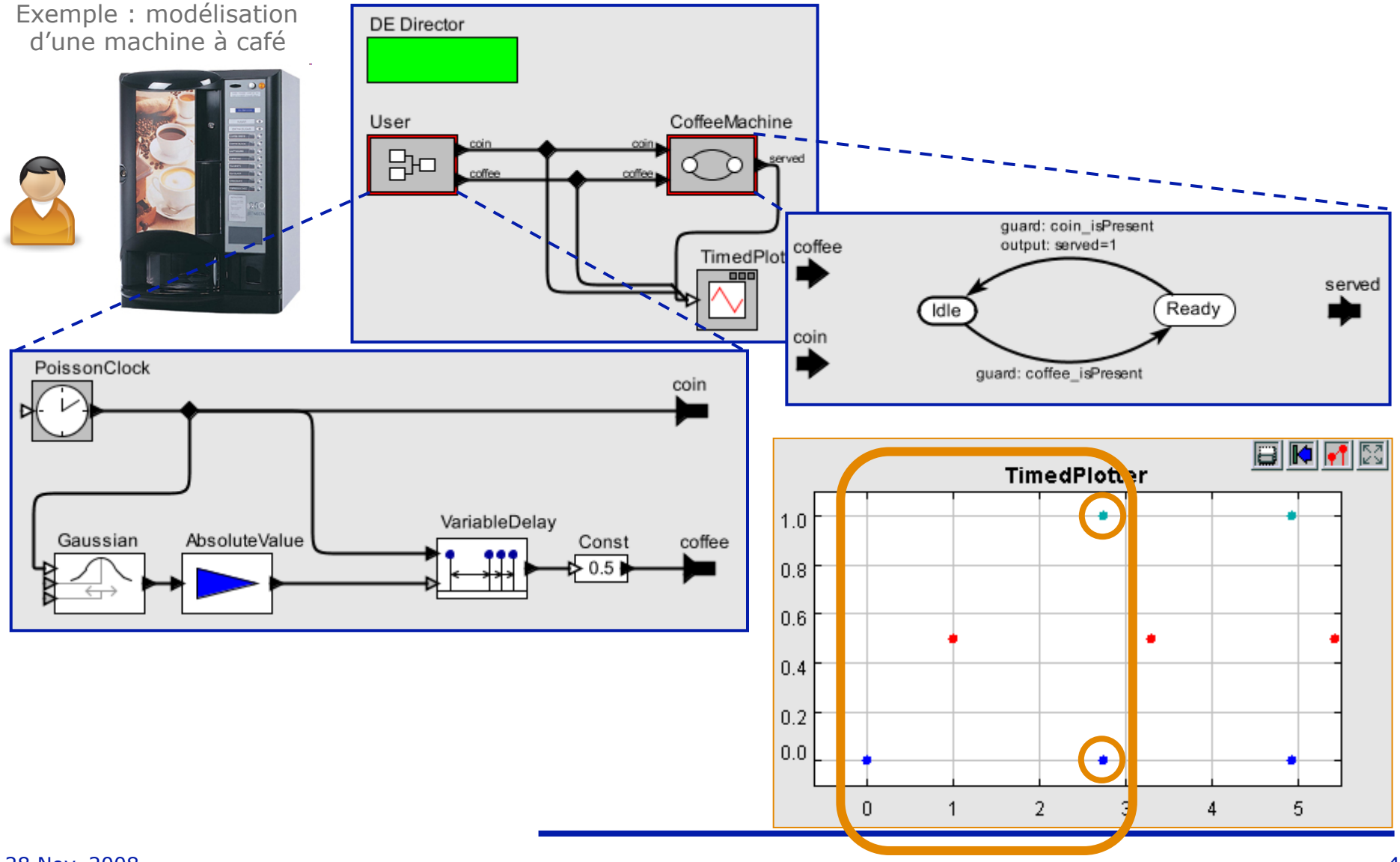
MoCs currently available

- Discrete Events (DE)
  - Exchange of events ⟨value, date⟩
  - ≈ Network messages
- Synchronous Data Flow (SDF)
  - Flows of sampled data
  - Multi sample rate
  - ≈ Simulink block diagrams
- Timed Finite State Machines (TFSM) [+ FSM + *Charts]
  - Timed transitions: "after(T)"
  - ≈ very simplified UML's Stateflow
- Petrinets

ModHel'X : positionnement par rapport à PtolemyII

Adaptation sémantique explicite entre MoCs

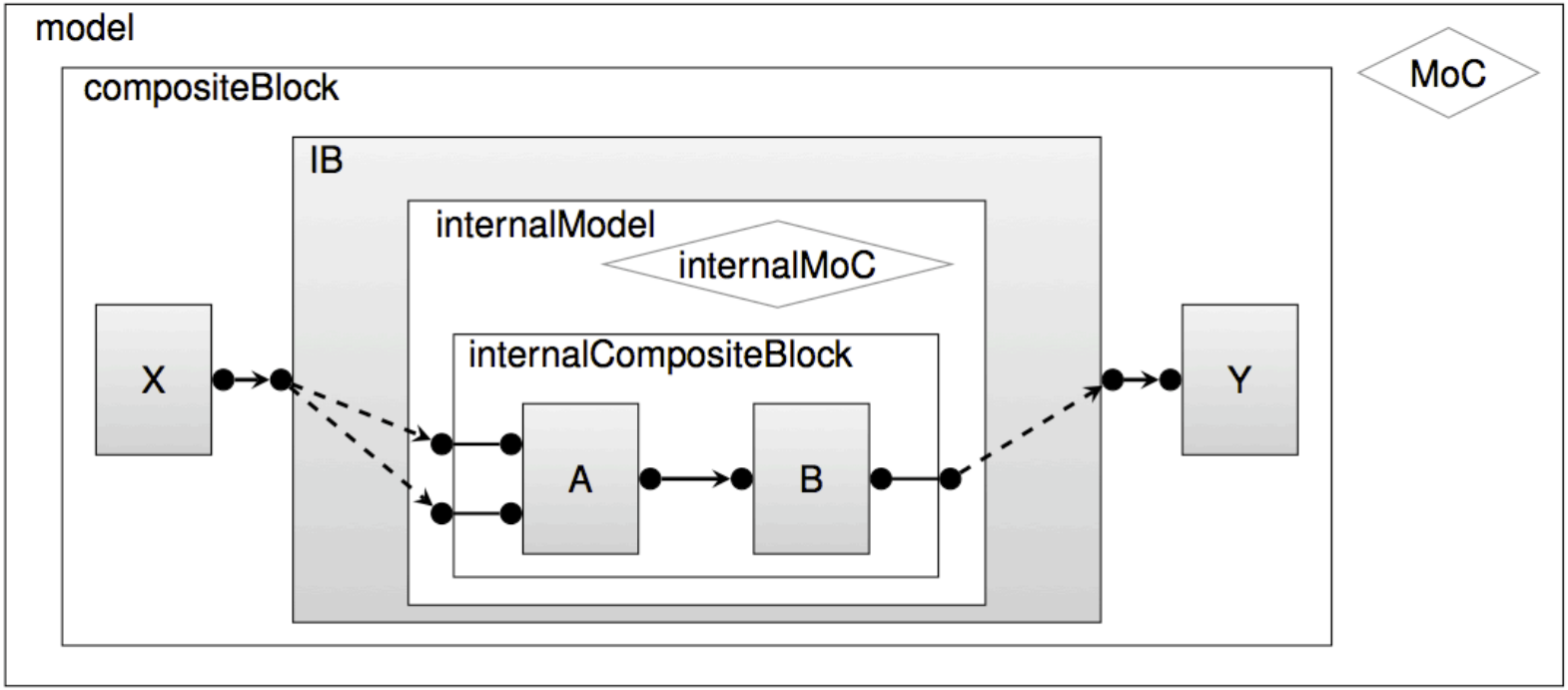
Exemple : modélisation d'une machine à café



Adaptation between MoCs

- Interface blocks are used to embed a model into a component

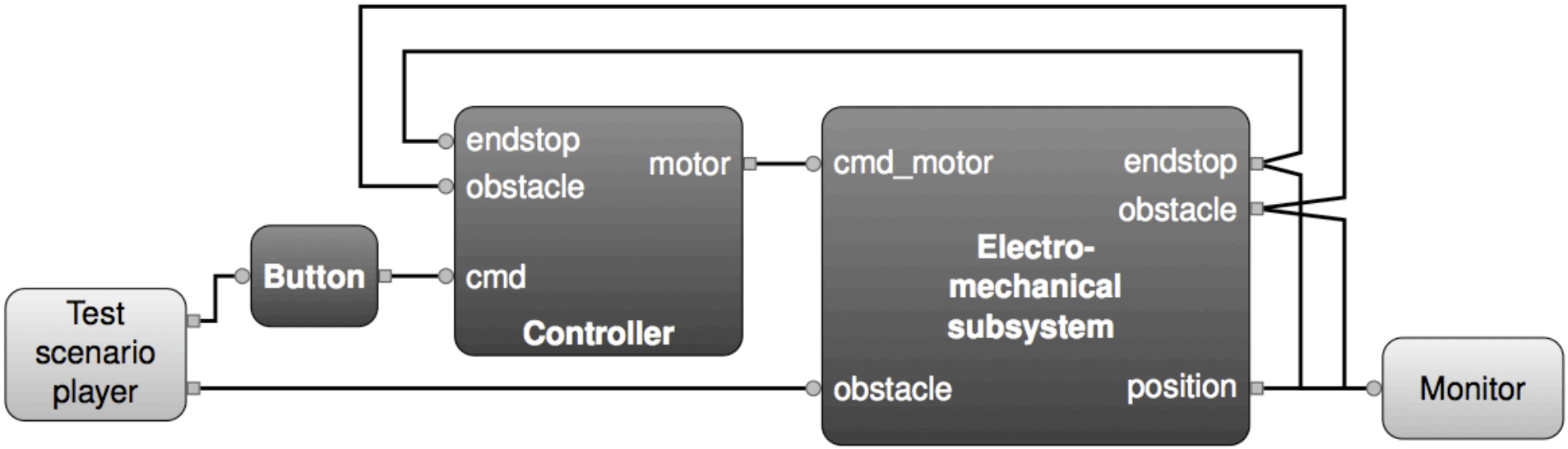
  → support for heterogeneity through hierarchy



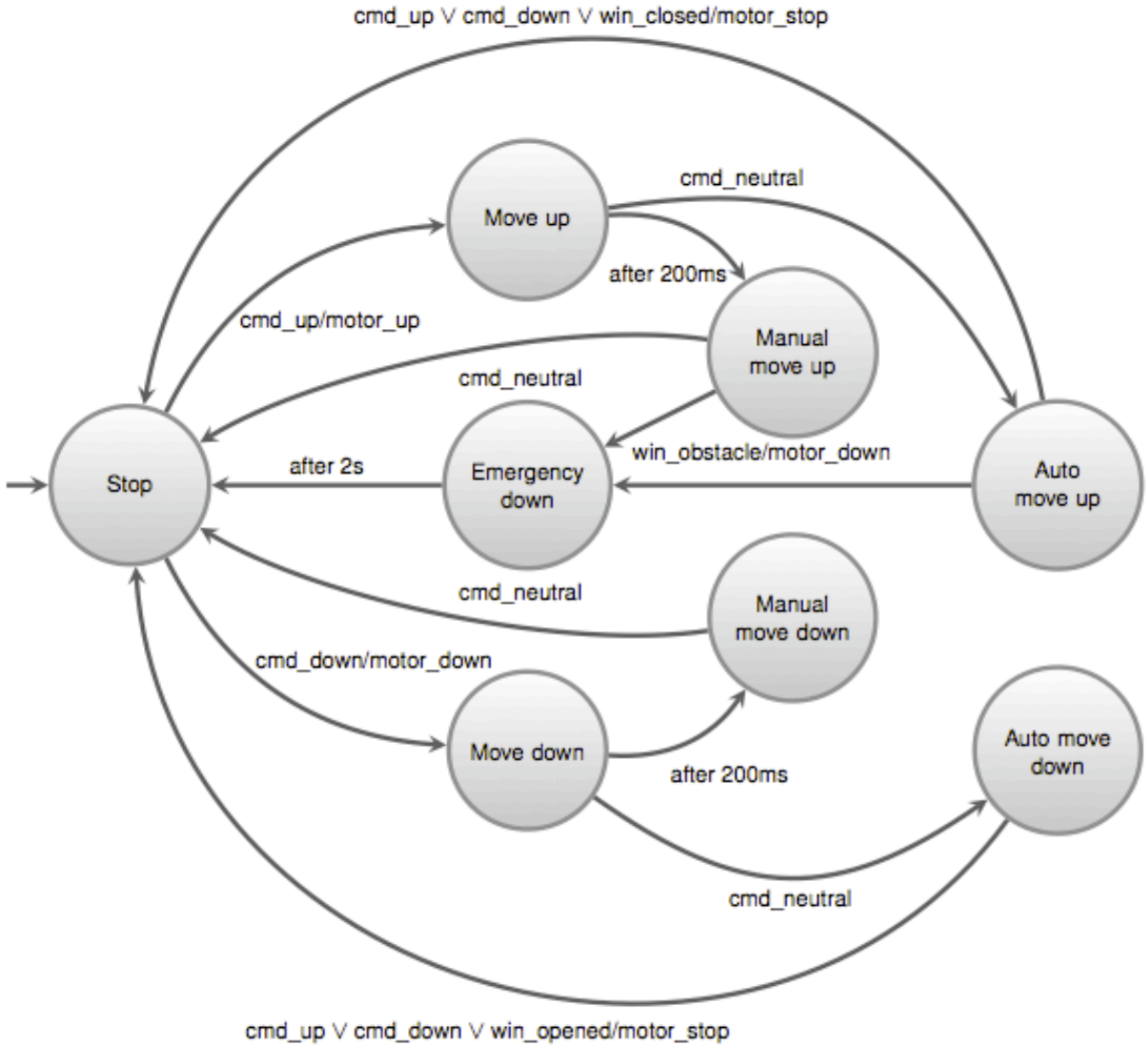
Contents

1. A few words on Supélec
2. Heterogeneous modeling
3. Basics of ModHel'X
4. Power Window example and demo

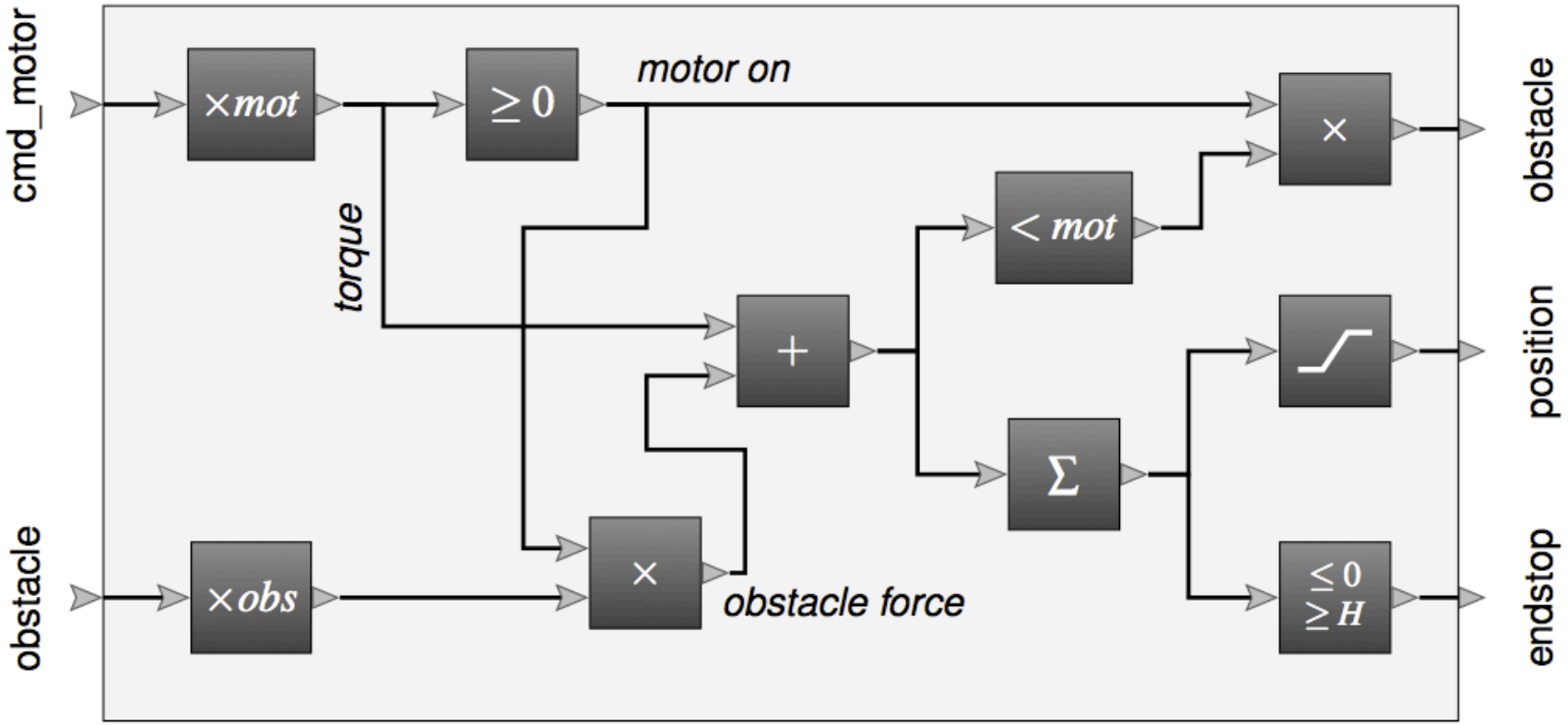
Example: power window Overview



Example: power window Controller (simplified)

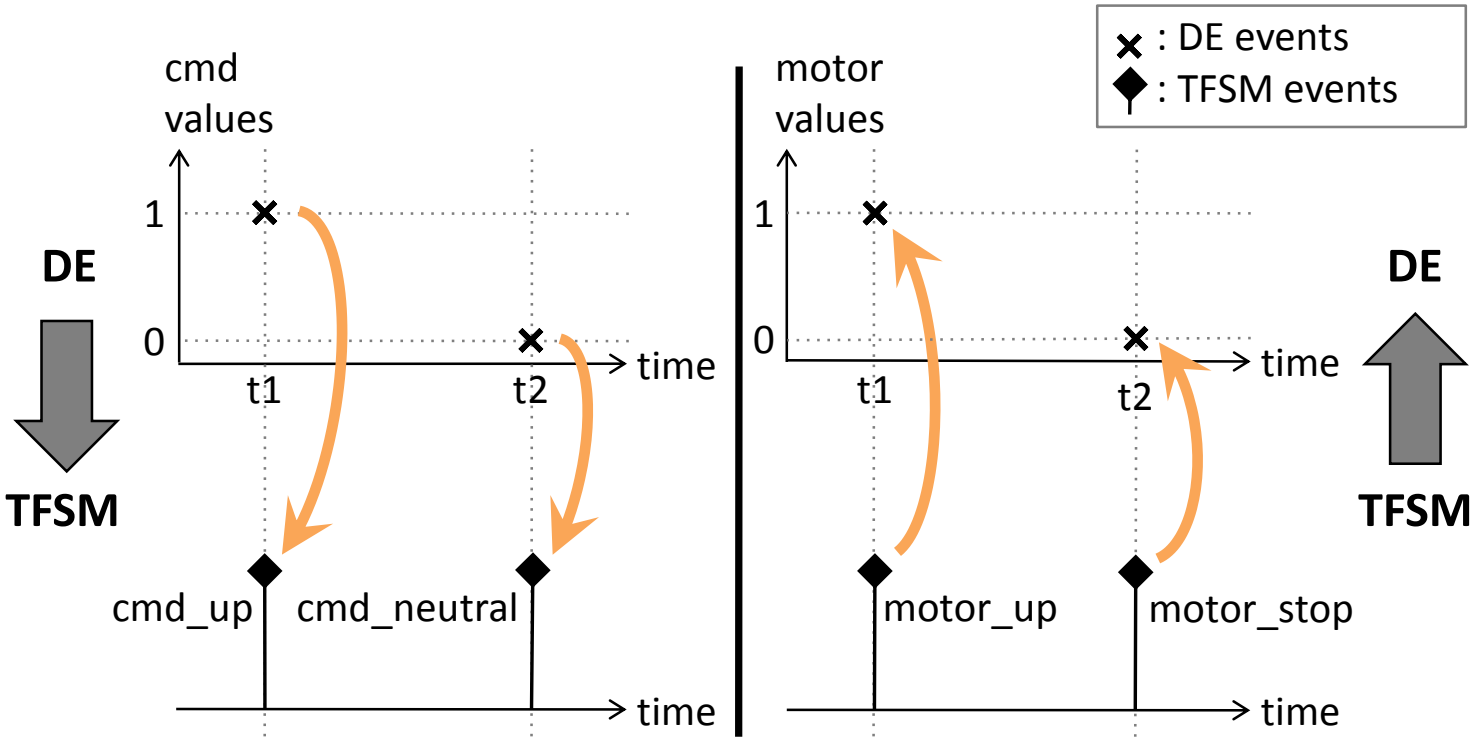


Example: power window
Physical subsystem (simplified)



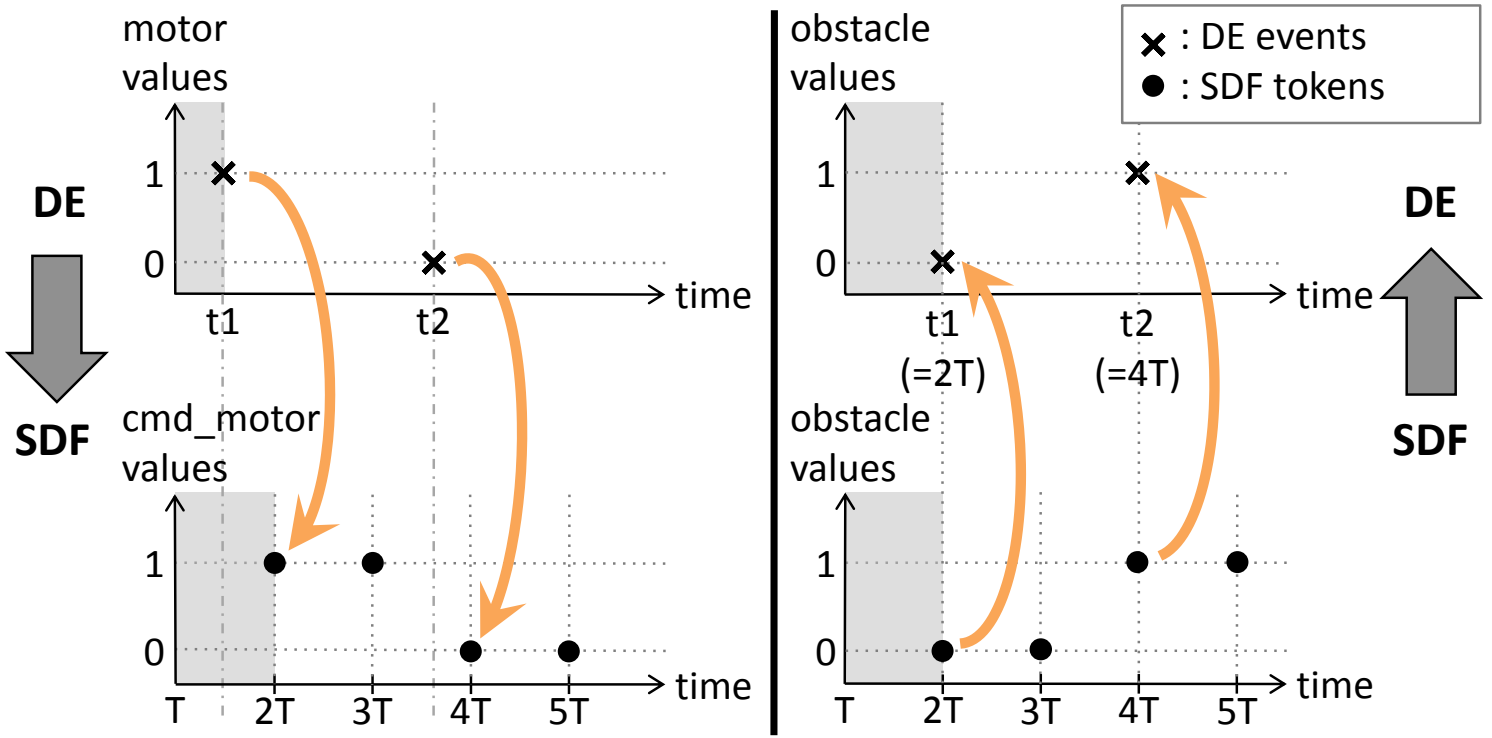
Example: power window DE/TFSM Adaptation

- Adaptation of
  - data
  - control
  - time



Example: power window DE/SDF Adaptation

- Adaptation of
  - data
  - control
  - time



Key points

- Explicit, re-usable description of
  - Models of Computation
  - Adaptation patterns between pairs of Models of Computation

- Originality of our approach
  - Extensible set of MoCs
    (≠ Stateflow/Simulink)
  - Completely explicit semantic adaptation
    (≠ VHDL-AMS)
  - Completely customizable semantic adaptation
    (≠ PtolemyII)