A process algebra for hybrid systems

Vashti Galpin
Laboratory for Foundations of Computer Science
University of Edinburgh

Joint work with Jane Hillston (University of Edinburgh)
and Luca Bortolussi (University of Trieste)

30 April 2008

Outline

Introduction

- hybrid systems

Introduction

- hybrid systems
  - discrete behaviour

Introduction

- hybrid systems
  - discrete behaviour
  - continuous behaviour, expressed as ODEs

Introduction

- hybrid systems
  - discrete behaviour
  - continuous behaviour, expressed as ODEs
- hybrid automata
  - well known
  - graphical rather than textual
  - not very compositional

Introduction

- hybrid systems
  - discrete behaviour
  - continuous behaviour, expressed as ODEs
- hybrid automata
  - well known
  - graphical rather than textual
  - not very compositional
- process algebras for hybrid systems
  - compositional language
  - semantic equivalences

Introduction (cont.)

- other process algebras

Introduction (cont.)

- other process algebras
  - ACP^{srt}_{hs} – Bergstra and Middelburg
  - HyPA – Cuijpers and Reniers
  - hybrid χ – van Beek *et al*
  - ϕ-calculus – Rounds and Song

Introduction (cont.)

- other process algebras
  - ACP^{srt}_{hs} – Bergstra and Middelburg
  - HyPA – Cuijpers and Reniers
  - hybrid χ – van Beek *et al*
  - ϕ-calculus – Rounds and Song
- comparison

Introduction (cont.)

- other process algebras
  - ACP^{srt}_{hs} – Bergstra and Middelburg
  - HyPA – Cuijpers and Reniers
  - hybrid χ – van Beek *et al*
  - ϕ-calculus – Rounds and Song
- comparison
  - differences: syntax, semantics, discontinuous behaviour, flow-determinism, theoretical results, tools – Khadim

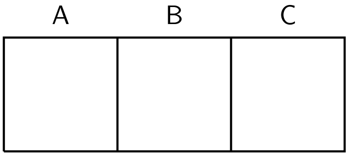
Introduction (cont.)

- other process algebras
  - ACP^{srt}_{hs} – Bergstra and Middelburg
  - HyPA – Cuijpers and Reniers
  - hybrid χ – van Beek *et al*
  - ϕ-calculus – Rounds and Song
- comparison
  - differences: syntax, semantics, discontinuous behaviour, flow-determinism, theoretical results, tools – Khadim
  - similarity: require full understanding of dynamic behaviour of subcomponents, ODEs appear in syntax

Introduction (cont.)

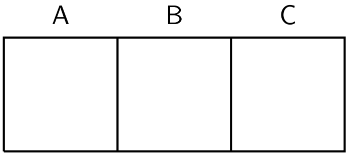
- other process algebras
  - ACP^{srt}_{hs} – Bergstra and Middelburg
  - HyPA – Cuijpers and Reniers
  - hybrid χ – van Beek *et al*
  - ϕ-calculus – Rounds and Song
- comparison
  - differences: syntax, semantics, discontinuous behaviour, flow-determinism, theoretical results, tools – Khadim
  - similarity: require full understanding of dynamic behaviour of subcomponents, ODEs appear in syntax
- HYPE
  - more fine-grained approach, individual additive flows
  - influence of continuous semantics of PEPA

Heater example



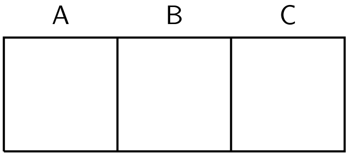
- three adjacent rooms

Heater example



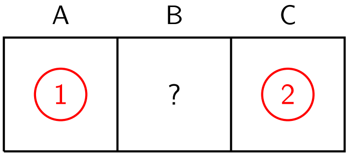
- three adjacent rooms
- fan heaters can be placed in each room
  - full effect on room
  - reduced effect on adjacent room
  - no effect on non-adjacent room

Heater example



- three adjacent rooms
- fan heaters can be placed in each room
  - full effect on room
  - reduced effect on adjacent room
  - no effect on non-adjacent room
- heaters can be switched on and off, max of 25°C

Heater example



- three adjacent rooms
- fan heaters can be placed in each room
  - full effect on room
  - reduced effect on adjacent room
  - no effect on non-adjacent room
- heaters can be switched on and off, max of 25°C
- how does the temperature in Room B change if there is one heater in Room A and one in Room C?

HYPE syntax

- two types of actions

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$\underline{\mathsf{a}} \in \mathcal{E}$

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$$\underline{\mathrm{a}} \in \mathcal{E}$$

- activities: influences on continuous aspect of system, flows

$$\alpha \in \mathcal{A} \qquad \alpha(\vec{X}) = (\iota, r, I(\vec{X}))$$

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$$\underline{\mathsf{a}} \in \mathcal{E}$$

- activities: influences on continuous aspect of system, flows

$$\alpha \in \mathcal{A} \qquad \alpha(\vec{X}) = (\iota, r, I(\vec{X}))$$

  - influence name $\iota \in IN$

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$$\underline{a} \in \mathcal{E}$$

- activities: influences on continuous aspect of system, flows

$$\alpha \in \mathcal{A} \qquad \alpha(\vec{X}) = (\iota, r, I(\vec{X}))$$

  - influence name $\iota \in IN$
  - rate $r \in \mathbb{R}$

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$$\underline{a} \in \mathcal{E}$$

- activities: influences on continuous aspect of system, flows

$$\alpha \in \mathcal{A} \qquad \alpha(\vec{X}) = (\iota, r, I(\vec{X}))$$

  - influence name $\iota \in IN$
  - rate $r \in \mathbb{R}$
  - influence type $I(\vec{X})$ with $[\![I(\vec{X})]\!] = f(\vec{X})$

HYPE syntax

- two types of actions
- events: instantaneous, discrete changes

$$\underline{a} \in \mathcal{E}$$

- activities: influences on continuous aspect of system, flows

$$\alpha \in \mathcal{A} \qquad \alpha(\vec{X}) = (\iota, r, I(\vec{X}))$$

  - influence name $\iota \in IN$
  - rate $r \in \mathbb{R}$
  - influence type $I(\vec{X})$ with $[\![I(\vec{X})]\!] = f(\vec{X})$
  - parameterised by formal variables $\vec{X}$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{\mathsf{a}} : \alpha.C_s \mid S + S \qquad \underline{\mathsf{a}} \in \mathcal{E}, \alpha \in \mathcal{A}$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{\mathsf{a}} : \alpha . C_s \mid S + S \qquad \underline{\mathsf{a}} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{a} : \alpha . C_s \mid S + S \qquad \underline{a} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$
- components: $P ::= C(\vec{X}) \mid P \underset{L}{\bowtie} P \qquad L \subseteq \mathcal{E}$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{a} : \alpha . C_s \mid S + S \qquad \underline{a} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$
- components: $P ::= C(\vec{X}) \mid P \underset{L}{\bowtie} P \qquad L \subseteq \mathcal{E}$
  - component names: $C(\vec{X}) \stackrel{def}{=} P$ or subcomponent name

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{a} : \alpha . C_s \mid S + S \qquad \underline{a} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$
- components: $P ::= C(\vec{X}) \mid P \underset{L}{\bowtie} P \qquad L \subseteq \mathcal{E}$
  - component names: $C(\vec{X}) \stackrel{def}{=} P$ or subcomponent name
- uncontrolled system: $\Sigma ::= C(\vec{V}) \mid \Sigma \underset{L}{\bowtie} \Sigma \qquad L \subseteq \mathcal{E}$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{a}:\alpha.C_s \mid S + S \qquad \underline{a} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$
- components: $P ::= C(\vec{X}) \mid P \underset{L}{\bowtie} P \qquad L \subseteq \mathcal{E}$
  - component names: $C(\vec{X}) \stackrel{def}{=} P$ or subcomponent name
- uncontrolled system: $\Sigma ::= C(\vec{V}) \mid \Sigma \underset{L}{\bowtie} \Sigma \qquad L \subseteq \mathcal{E}$
- controller: $M ::= \underline{a}.M \mid 0 \mid M + M \qquad \underline{a} \in \mathcal{E}$

  $Con ::= M \mid Con \underset{L}{\bowtie} Con \qquad L \subseteq \mathcal{E}$

HYPE syntax (cont.)

- subcomponents: $S ::= \underline{a}:\alpha.C_s \mid S + S \qquad \underline{a} \in \mathcal{E}, \alpha \in \mathcal{A}$
  - subcomponent names: $C_s(\vec{X}) \stackrel{def}{=} S$
- components: $P ::= C(\vec{X}) \mid P \bowtie_L P \qquad L \subseteq \mathcal{E}$
  - component names: $C(\vec{X}) \stackrel{def}{=} P$ or subcomponent name
- uncontrolled system: $\Sigma ::= C(\vec{V}) \mid \Sigma \bowtie_L \Sigma \qquad L \subseteq \mathcal{E}$
- controller: $M ::= \underline{a}.M \mid 0 \mid M + M \qquad \underline{a} \in \mathcal{E}$
  $Con ::= M \mid Con \bowtie_L Con \qquad L \subseteq \mathcal{E}$
- controlled system: $ConSys ::= \Sigma \bowtie_L \underline{\mathrm{init}}.Con \qquad L \subseteq \mathcal{E}$

HYPE syntax (cont.)

- HYPE model: $(\mathit{ConSys}, \mathcal{V}, \mathcal{X}, \mathit{IN}, \mathit{IT}, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, \mathit{EC}, \mathit{ID})$

HYPE syntax (cont.)

- HYPE model: $(\mathit{ConSys}, \mathcal{V}, \mathcal{X}, \mathit{IN}, \mathit{IT}, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, \mathit{EC}, \mathit{ID})$
  - *IN* influence names, *IT* influence types

HYPE syntax (cont.)

- HYPE model: $(\mathit{ConSys}, \mathcal{V}, \mathcal{X}, \mathit{IN}, \mathit{IT}, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, \mathit{EC}, \mathit{ID})$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \to \mathit{EC}$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)

HYPE syntax (cont.)

- HYPE model: $(ConSys, \mathcal{V}, \mathcal{X}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \to EC$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : IN \to \mathcal{V}$, association of influence names with variables

HYPE syntax (cont.)

- HYPE model: $(ConSys, \mathcal{V}, \mathcal{X}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \rightarrow EC$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : IN \rightarrow \mathcal{V}$, association of influence names with variables
  - ID influence descriptions, $[\![I(\vec{X})]\!] = f(\vec{X})$,

HYPE syntax (cont.)

- HYPE model: $(ConSys, \mathcal{V}, \mathcal{X}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \rightarrow EC$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : IN \rightarrow \mathcal{V}$, association of influence names with variables
  - ID influence descriptions, $[\![I(\vec{X})]\!] = f(\vec{X})$,
- well-defined HYPE model

HYPE syntax (cont.)

- HYPE model: $(\mathit{ConSys}, \mathcal{V}, \mathcal{X}, \mathit{IN}, \mathit{IT}, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, \mathit{EC}, \mathit{ID})$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \to \mathit{EC}$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : \mathit{IN} \to \mathcal{V}$, association of influence names with variables
  - ID influence descriptions, $[\![I(\vec{X})]\!] = f(\vec{X})$,
- well-defined HYPE model
  - subcomponents:

$$\mathcal{C}_s(\vec{X}) \stackrel{\mathit{def}}{=} \underline{\mathsf{a}}_1 : \alpha_1.\mathcal{C}_s(\vec{X}) + \ldots + \underline{\mathsf{a}}_n : \alpha_n.\mathcal{C}_s(\vec{X}) \quad \underline{\mathsf{a}}_i \neq \underline{\mathsf{a}}_j$$

HYPE syntax (cont.)

- HYPE model: $(\mathit{ConSys}, \mathcal{V}, \mathcal{X}, \mathit{IN}, \mathit{IT}, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, \mathit{EC}, \mathit{ID})$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \to \mathit{EC}$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : \mathit{IN} \to \mathcal{V}$, association of influence names with variables
  - ID influence descriptions, $[\![I(\vec{X})]\!] = f(\vec{X})$,
- well-defined HYPE model
  - subcomponents:
    $$\mathcal{C}_s(\vec{X}) \stackrel{def}{=} \underline{a}_1 : \alpha_1.\mathcal{C}_s(\vec{X}) + \ldots + \underline{a}_n : \alpha_n.\mathcal{C}_s(\vec{X}) \quad \underline{a}_i \neq \underline{a}_j$$
  - $\underline{\text{init}} : (\iota, \_, \_)$ appears exactly once
  - $\underline{a} : (\iota, \_, \_)$ appears at most once

HYPE syntax (cont.)

- HYPE model: $(ConSys, \mathcal{V}, \mathcal{X}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$
  - IN influence names, IT influence types
  - $\mathrm{ec} : \mathcal{E} \rightarrow EC$, association of events with event conditions
  - EC, event conditions, (activation condition, reset)
  - $\mathrm{iv} : IN \rightarrow \mathcal{V}$, association of influence names with variables
  - ID influence descriptions, $[\![I(\vec{X})]\!] = f(\vec{X})$,
- well-defined HYPE model
  - subcomponents:
    $$\mathcal{C}_s(\vec{X}) \stackrel{def}{=} \underline{a}_1 : \alpha_1.\mathcal{C}_s(\vec{X}) + \ldots + \underline{a}_n : \alpha_n.\mathcal{C}_s(\vec{X}) \quad \underline{a}_i \neq \underline{a}_j$$
  - $\underline{\mathrm{init}} : (\iota, \_, \_)$ appears exactly once
  - $\underline{a} : (\iota, \_, \_)$ appears at most once
  - synchronisation on shared events

Heater example (cont.)

- room: $Room_x(T) \stackrel{def}{=} \underline{\text{init}}{:}(t_{0,x}, -1, linear(T)).Room_x(T)$
  - $t_{0,x}$ represents influence of cooling on Room x

Heater example (cont.)

- room: $Room_x(T) \stackrel{def}{=} \underline{\text{init}} : (t_{0,x}, -1, linear(T)).Room_x(T)$
  - $t_{0,x}$ represents influence of cooling on Room x
- fan i in Room x affecting Room y:
  - $t_{i,y}$ represents influence of fan i on Room y

Heater example (cont.)

- room: $Room_x(T) \stackrel{def}{=} \underline{\text{init}} : (t_{0,x}, -1, linear(T)).Room_x(T)$
  - $t_{0,x}$ represents influence of cooling on Room x
- fan i in Room x affecting Room y:
  - $t_{i,y}$ represents influence of fan i on Room y

$$\begin{aligned} Fan_{i,x,y} &\stackrel{def}{=} \underline{\text{init}} : (t_{i,y}, 0, const).Fan_{i,x,y} + \\ &\quad \underline{\text{off}}_i : (t_{i,y}, 0, const).Fan_{i,x,y} + \\ &\quad \underline{\text{on}}_i : (t_{i,y}, r_i, const_{\psi(x,y)}).Fan_{i,x,y} \end{aligned}$$

where

$$\psi(x,y) = \begin{cases} in & \text{if } x = y \\ adj & \text{if } x \text{ and } y \text{ are adjacent} \\ far & \text{otherwise} \end{cases}$$

Heater example (cont.)

- uncontrolled system:

$$Sys \stackrel{def}{=} (Fan_{1,A,B} \underset{\{\underline{init}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{init}\}}{\bowtie} Room_B(T_B)$$

Heater example (cont.)

- uncontrolled system:

$$Sys \stackrel{def}{=} (Fan_{1,A,B} \underset{\{\underline{\text{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\text{init}}\}}{\bowtie} Room_B(T_B)$$

- controller:

$$Con \stackrel{def}{=} Con_1 \underset{\emptyset}{\bowtie} Con_2 \qquad Con_i \stackrel{def}{=} \underline{\text{on}}_i.\underline{\text{off}}_i.Con_i$$

Heater example (cont.)

- uncontrolled system:

$$Sys \stackrel{def}{=} (Fan_{1,A,B} \underset{\{\underline{\text{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\text{init}}\}}{\bowtie} Room_B(T_B)$$

- controller:

$$Con \stackrel{def}{=} Con_1 \underset{\emptyset}{\bowtie} Con_2 \qquad Con_i \stackrel{def}{=} \underline{\text{on}}_i.\underline{\text{off}}_i.Con_i$$

- controlled system:

$$MF \stackrel{def}{=} Sys \underset{M}{\bowtie} \underline{\text{init}}.Con \qquad M = \{\underline{\text{init}}, \underline{\text{on}}_1, \underline{\text{off}}_1, \underline{\text{on}}_2, \underline{\text{off}}_2\}$$

Heater example (cont.)

- uncontrolled system:

$$Sys \stackrel{def}{=} (Fan_{1,A,B} \underset{\{\underline{\text{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\text{init}}\}}{\bowtie} Room_B(T_B)$$

- controller:

$$Con \stackrel{def}{=} Con_1 \underset{\emptyset}{\bowtie} Con_2 \qquad Con_i \stackrel{def}{=} \underline{\text{on}}_i.\underline{\text{off}}_i.Con_i$$

- controlled system:

$$MF \stackrel{def}{=} Sys \underset{M}{\bowtie} \underline{\text{init}}.Con \qquad M = \{\underline{\text{init}}, \underline{\text{on}}_1, \underline{\text{off}}_1, \underline{\text{on}}_2, \underline{\text{off}}_2\}$$

- MF is well-defined

Heater example (cont.)

- $(MF, \{T_B\}, \{T\}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$

$$\begin{array}{rcl} IN & = & \{t_{0,B}, t_{1,B}, t_{2,B}\} \\ IT & = & \{const, const_{in}, const_{adj}, const_{far}, linear(T)\} \\ \mathcal{E} & = & \{\underline{\mathrm{init}}, \underline{\mathrm{on}}_1, \underline{\mathrm{off}}_1, \underline{\mathrm{on}}_2, \underline{\mathrm{off}}_2\} \end{array}$$

Heater example (cont.)

- $(MF, \{T_B\}, \{T\}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$

$$\begin{aligned} IN &= \{t_{0,B}, t_{1,B}, t_{2,B}\} \\ IT &= \{const, const_{in}, const_{adj}, const_{far}, linear(T)\} \\ \mathcal{E} &= \{\underline{\text{init}}, \underline{\text{on}}_1, \underline{\text{off}}_1, \underline{\text{on}}_2, \underline{\text{off}}_2\} \end{aligned}$$

$$\begin{aligned} \mathrm{ec}(\underline{\text{init}}) &= (true, (T'_B = T_0)) \\ \mathrm{ec}(\underline{\text{on}}_i) &= (\bot, (T'_B = T_B)) \\ \mathrm{ec}(\underline{\text{off}}_i) &= ((T_B = 25), (T'_B = T_B)) \end{aligned}$$

Heater example (cont.)

- $(MF, \{T_B\}, \{T\}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$

$$\begin{array}{rcl} IN & = & \{t_{0,B}, t_{1,B}, t_{2,B}\} \\ IT & = & \{const, const_{in}, const_{adj}, const_{far}, linear(T)\} \\ \mathcal{E} & = & \{\underline{\text{init}}, \underline{\text{on}}_1, \underline{\text{off}}_1, \underline{\text{on}}_2, \underline{\text{off}}_2\} \end{array}$$

$$\begin{array}{rcl} \mathrm{ec}(\underline{\text{init}}) & = & (true, (T_B' = T_0)) \\ \mathrm{ec}(\underline{\text{on}}_i) & = & (\bot, (T_B' = T_B)) \\ \mathrm{ec}(\underline{\text{off}}_i) & = & ((T_B = 25), (T_B' = T_B)) \end{array}$$

$$\mathrm{iv}(t_{i,B}) = T_B$$

Heater example (cont.)

- $(MF, \{T_B\}, \{T\}, IN, IT, \mathcal{E}, \mathcal{A}, \mathrm{ec}, \mathrm{iv}, EC, ID)$

$$
\begin{aligned}
IN &= \{t_{0,B}, t_{1,B}, t_{2,B}\} \\
IT &= \{const, const_{in}, const_{adj}, const_{far}, linear(T)\} \\
\mathcal{E} &= \{\underline{\mathrm{init}}, \underline{\mathrm{on}}_1, \underline{\mathrm{off}}_1, \underline{\mathrm{on}}_2, \underline{\mathrm{off}}_2\}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{ec}(\underline{\mathrm{init}}) &= (true, (T_B' = T_0)) \\
\mathrm{ec}(\underline{\mathrm{on}}_i) &= (\bot, (T_B' = T_B)) \\
\mathrm{ec}(\underline{\mathrm{off}}_i) &= ((T_B = 25), (T_B' = T_B))
\end{aligned}
$$

$$
\mathrm{iv}(t_{i,B}) = T_B
$$

$$
\begin{aligned}
[\![const]\!] &= 0 \qquad [\![linear(X)]\!] = X \\
[\![const_{in}]\!] &= 1 \qquad [\![const_{adj}]\!] = 0.5 \qquad [\![const_{far}]\!] = 0
\end{aligned}
$$

Operational semantics

- state: $\sigma : \mathit{IN} \longrightarrow (\mathbb{R} \times \mathit{IT})$

Operational semantics

- state: $\sigma : \mathit{IN} \rightarrow (\mathbb{R} \times \mathit{IT})$
- configuration: $\langle \mathit{ConSys}, \sigma \rangle$

Operational semantics

- state: $\sigma : \mathit{IN} \rightarrow (\mathbb{R} \times \mathit{IT})$
- configuration: $\langle \mathit{ConSys}, \sigma \rangle$
- labelled transition system: $(\mathcal{F}, \mathcal{E}, \rightarrow \subseteq \mathcal{F} \times \mathcal{E} \times \mathcal{F})$

Operational semantics

- state: $\sigma : IN \rightarrow (\mathbb{R} \times IT)$
- configuration: $\langle ConSys, \sigma \rangle$
- labelled transition system: $(\mathcal{F}, \mathcal{E}, \rightarrow \subseteq \mathcal{F} \times \mathcal{E} \times \mathcal{F})$
- updating function: $\sigma[\iota \mapsto (r, I)]$

$$\sigma[\iota \mapsto (r, I)](x) = \begin{cases} (r, I) & \text{if } x = \iota \\ \sigma(x) & \text{otherwise} \end{cases}$$

Operational semantics

- state: $\sigma : IN \rightarrow (\mathbb{R} \times IT)$
- configuration: $\langle ConSys, \sigma \rangle$
- labelled transition system: $(\mathcal{F}, \mathcal{E}, \rightarrow \subseteq \mathcal{F} \times \mathcal{E} \times \mathcal{F})$
- updating function: $\sigma[\iota \mapsto (r, I)]$

$$\sigma[\iota \mapsto (r, I)](x) = \begin{cases} (r, I) & \text{if } x = \iota \\ \sigma(x) & \text{otherwise} \end{cases}$$

- change identifying function: $\Gamma : \mathcal{S} \times \mathcal{S} \times \mathcal{S} \rightarrow \mathcal{S}$

$$(\Gamma(\sigma, \tau, \tau'))(\iota) = \begin{cases} \tau(\iota) & \text{if } \sigma(\iota) = \tau'(\iota) \\ \tau'(\iota) & \text{if } \sigma(\iota) = \tau(\iota) \\ \text{undefined} & \text{otherwise} \end{cases}$$

Operational semantics (cont.)

Prefix with influence:

$$\frac{}{\langle \underline{a} : (\iota, r, I).E, \sigma \rangle \xrightarrow{\underline{a}} \langle E, \sigma[\iota \mapsto (r, I)] \rangle}$$

Prefix without influence:

$$\frac{}{\langle \underline{a}.E, \sigma \rangle \xrightarrow{\underline{a}} \langle E, \sigma \rangle}$$

Choice:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle}{\langle E + F, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle} \qquad \frac{\langle F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \sigma' \rangle}{\langle E + F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \sigma' \rangle}$$

Constant:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle}{\langle A, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle} (A \stackrel{def}{=} E)$$

Operational semantics (cont.)

Prefix with influence:

$$\frac{}{\langle \underline{a}:(\iota, r, I).E, \sigma\rangle \xrightarrow{\underline{a}} \langle E, \sigma[\iota \mapsto (r, I)]\rangle}$$

Prefix without influence:

$$\frac{}{\langle \underline{a}.E, \sigma\rangle \xrightarrow{\underline{a}} \langle E, \sigma\rangle}$$

Choice:

$$\frac{\langle E, \sigma\rangle \xrightarrow{\underline{a}} \langle E', \sigma'\rangle}{\langle E + F, \sigma\rangle \xrightarrow{\underline{a}} \langle E', \sigma'\rangle} \qquad \frac{\langle F, \sigma\rangle \xrightarrow{\underline{a}} \langle F', \sigma'\rangle}{\langle E + F, \sigma\rangle \xrightarrow{\underline{a}} \langle F', \sigma'\rangle}$$

Constant:

$$\frac{\langle E, \sigma\rangle \xrightarrow{\underline{a}} \langle E', \sigma'\rangle}{\langle A, \sigma\rangle \xrightarrow{\underline{a}} \langle E', \sigma'\rangle}(A \stackrel{def}{=} E)$$

Operational semantics (cont.)

Parallel without synchronisation:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E' \underset{M}{\bowtie} F, \sigma' \rangle} \quad \underline{a} \notin M$$

$$\frac{\langle F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \sigma' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E \underset{M}{\bowtie} F', \sigma' \rangle} \quad \underline{a} \notin M$$

Parallel with synchronisation:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \tau \rangle \quad \langle F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \tau' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E' \underset{M}{\bowtie} F', \Gamma(\sigma, \tau, \tau') \rangle}$$

$\underline{a} \in M, \Gamma$ defined

Operational semantics (cont.)

Parallel without synchronisation:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \sigma' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E' \underset{M}{\bowtie} F, \sigma' \rangle} \quad \underline{a} \notin M$$

$$\frac{\langle F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \sigma' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E \underset{M}{\bowtie} F', \sigma' \rangle} \quad \underline{a} \notin M$$

Parallel with synchronisation:

$$\frac{\langle E, \sigma \rangle \xrightarrow{\underline{a}} \langle E', \tau \rangle \quad \langle F, \sigma \rangle \xrightarrow{\underline{a}} \langle F', \tau' \rangle}{\langle E \underset{M}{\bowtie} F, \sigma \rangle \xrightarrow{\underline{a}} \langle E' \underset{M}{\bowtie} F', \Gamma(\sigma, \tau, \tau') \rangle}$$

$\underline{a} \in M, \Gamma$ defined

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B}, \tau_1 \rangle \qquad \langle F_{2,C,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{2,C,B}, \tau_2 \rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3 \rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \textcolor{red}{\tau} \rangle \xrightarrow{\text{init}} \langle F_{1,A,B}, \tau_1 \rangle \qquad \langle F_{2,C,B}, \textcolor{red}{\tau} \rangle \xrightarrow{\text{init}} \langle F_{2,C,B}, \tau_2 \rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \textcolor{red}{\tau} \rangle \xrightarrow{\text{init}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3 \rangle}$$

$$\begin{aligned}
\textcolor{red}{\tau} &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \textcolor{red}{\tau}[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \textcolor{red}{\tau}[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\textcolor{red}{\tau}, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \tau \rangle \xrightarrow{\text{init}} \langle F_{1,A,B}, \tau_1 \rangle \qquad \langle F_{2,C,B}, \tau \rangle \xrightarrow{\text{init}} \langle F_{2,C,B}, \tau_2 \rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau \rangle \xrightarrow{\text{init}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3 \rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{1,A,B}, \tau_1\rangle \qquad \langle F_{2,C,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{2,C,B}, \tau_2\rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3\rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \tau\rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B}, \tau_1\rangle \qquad \langle F_{2,C,B}, \tau\rangle \xrightarrow{\underline{\text{init}}} \langle F_{2,C,B}, \tau_2\rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau\rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3\rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

- transition derivation

$$\frac{\langle F_{1,A,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B}, \tau_1 \rangle \qquad \langle F_{2,C,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{2,C,B}, \tau_2 \rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau \rangle \xrightarrow{\underline{\text{init}}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3 \rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

Heater example (cont.)

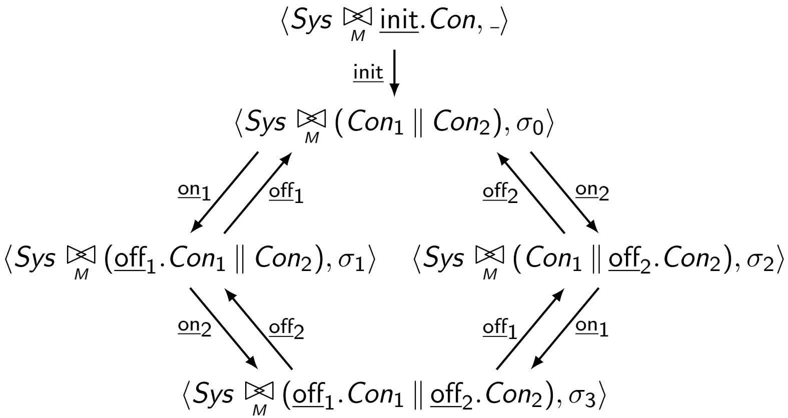
- transition derivation

$$\frac{\langle F_{1,A,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{1,A,B}, \tau_1\rangle \qquad \langle F_{2,C,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{2,C,B}, \tau_2\rangle}{\langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau\rangle \xrightarrow{\text{init}} \langle F_{1,A,B} \underset{\underline{\text{init}}}{\bowtie} F_{2,C,B}, \tau_3\rangle}$$

$$\begin{aligned}
\tau &= \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto *\} \\
\tau_1 &= \tau[t_{1,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto *\} \\
\tau_2 &= \tau[t_{2,B} \mapsto (0,c)] = \{t_{0,B} \mapsto *, t_{1,B} \mapsto *, t_{2,B} \mapsto (0,c)\} \\
\tau_3 &= \Gamma(\tau, \tau_1, \tau_2) = \{t_{0,B} \mapsto *, t_{1,B} \mapsto (0,c), t_{2,B} \mapsto (0,c)\}
\end{aligned}$$

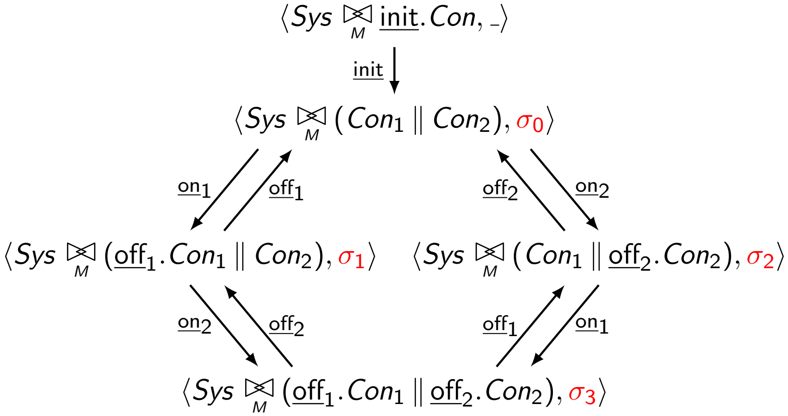
Heater example (cont.)

- labelled transition system



Heater example (cont.)

- labelled transition system



Heater example (cont.)

- for MF, there are four states

$$
\begin{aligned}
\sigma_0 &= \{t_{0,B} \mapsto (-1, linear(T_B)), t_{1,B} \mapsto (0, const), \\
&\qquad t_{2,B} \mapsto (0, const)\} \\
\sigma_1 &= \{t_{0,B} \mapsto (-1, linear(T_B)), t_{1,B} \mapsto (r_1, const_{adj}), \\
&\qquad t_{2,B} \mapsto (0, const)\} \\
\sigma_2 &= \{t_{0,B} \mapsto (-1, linear(T_B)), t_{1,B} \mapsto (0, const), \\
&\qquad t_{2,B} \mapsto (r_2, const_{adj})\} \\
\sigma_3 &= \{t_{0,B} \mapsto (-1, linear(T_B)), t_{1,B} \mapsto (r_1, const_{adj}), \\
&\qquad t_{2,B} \mapsto (r_2, const_{adj})\}
\end{aligned}
$$

Hybrid semantics

- extract ODEs from each state σ in the lts of CS

$$CS_\sigma = \left\{ \text{ODE for variable } V \;\middle|\; V \in \mathcal{V} \right\} \text{ where}$$

$$\frac{dV}{dt} = \sum \left\{ r[\![I(\vec{W})]\!] \;\middle|\; \mathrm{iv}(\iota) = V \text{ and } \sigma(\iota) = (r, I(\vec{W})) \right\}$$

Hybrid semantics

- extract ODEs from each state σ in the lts of CS

$$CS_\sigma = \left\{ \text{ODE for variable } V \;\middle|\; V \in \mathcal{V} \right\} \text{ where}$$

$$\frac{dV}{dt} = \sum \left\{ r[\![I(\vec{W})]\!] \;\middle|\; \mathrm{iv}(\iota) = V \text{ and } \sigma(\iota) = (r, I(\vec{W})) \right\}$$

  - for any influence name associated with V

Hybrid semantics

- extract ODEs from each state σ in the lts of CS

$$CS_\sigma = \left\{ \text{ODE for variable } V \;\middle|\; V \in \mathcal{V} \right\} \text{ where}$$

$$\frac{dV}{dt} = \sum \left\{ r[\![I(\vec{W})]\!] \;\middle|\; \mathrm{iv}(\iota) = V \text{ and } \sigma(\iota) = (r, I(\vec{W})) \right\}$$

  - for any influence name associated with V
  - determine from σ its rate and influence type

Hybrid semantics

- extract ODEs from each state σ in the lts of CS

$$CS_\sigma = \left\{ \text{ODE for variable } V \;\middle|\; V \in \mathcal{V} \right\} \text{ where}$$

$$\frac{dV}{dt} = \sum \left\{ r[\![I(\vec{W})]\!] \;\middle|\; \mathrm{iv}(\iota) = V \text{ and } \sigma(\iota) = (r, I(\vec{W})) \right\}$$

  - for any influence name associated with V
  - determine from σ its rate and influence type
  - multiply its rate and influence function together

Hybrid semantics

- extract ODEs from each state σ in the lts of CS

$$CS_\sigma = \left\{ \text{ODE for variable } V \;\middle|\; V \in \mathcal{V} \right\} \text{ where}$$

$$\frac{dV}{dt} = \sum \left\{ r[\![I(\vec{W})]\!] \;\middle|\; \mathrm{iv}(\iota) = V \text{ and } \sigma(\iota) = (r, I(\vec{W})) \right\}$$

  - for any influence name associated with V
  - determine from σ its rate and influence type
  - multiply its rate and influence function together
  - sum these over all associated influence names

Heater example (cont.)

- state σ_0 occurs when both fans are off

$$MF_{\sigma_0} = \left\{ \frac{dT_B}{dt} = -T_B \right\}$$

Heater example (cont.)

- state σ_0 occurs when both fans are off

$$MF_{\sigma_0} = \left\{ \frac{dT_B}{dt} = -T_B \right\}$$

- state σ_1 occurs when fan 1 is on

$$MF_{\sigma_1} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5r_1 \right\}$$

Heater example (cont.)

- state σ_0 occurs when both fans are off

$$MF_{\sigma_0} = \left\{ \frac{dT_B}{dt} = -T_B \right\}$$

- state σ_1 occurs when fan 1 is on

$$MF_{\sigma_1} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5r_1 \right\}$$

- state σ_2 occurs when fan 2 is on

$$MF_{\sigma_2} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5r_2 \right\}$$

Heater example (cont.)

- state σ_0 occurs when both fans are off

$$MF_{\sigma_0} = \left\{ \frac{dT_B}{dt} = -T_B \right\}$$

- state σ_1 occurs when fan 1 is on

$$MF_{\sigma_1} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5r_1 \right\}$$

- state σ_2 occurs when fan 2 is on

$$MF_{\sigma_2} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5r_2 \right\}$$

- state σ_3 occurs when both fans are on

$$MF_{\sigma_3} = \left\{ \frac{dT_B}{dt} = -T_B + 0.5(r_1 + r_2) \right\}$$

Heater example in ACP^{srt}_{hs}

- system with temperature limit
- $\theta \equiv (T_B^{\bullet} = {}^{\bullet}T_B) \quad \psi \equiv (T_B = 25)$

$$\begin{aligned}
\text{Start} &\stackrel{def}{=} (T_B = T_0) \curlywedge \text{Off12} \\
\text{Off12} &\stackrel{def}{=} (\dot{T}_B = -T_B) \curvearrowright \sigma^*_{\text{rel}}(\theta \ulcorner (on_1 \cdot \text{On1} + on_2 \cdot \text{On2})) \\
\text{On1} &\stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5r_1) \\
&\quad \curvearrowright \sigma^*_{\text{rel}}((\theta \ulcorner on_2 \cdot \text{On12}) + (\psi :\longrightarrow (\theta \ulcorner off_1 \cdot \text{Off12}))) \\
\text{On2} &\stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5r_2) \\
&\quad \curvearrowright \sigma^*_{\text{rel}}((\theta \ulcorner on_1 \cdot \text{On12}) + (\psi :\longrightarrow (\theta \ulcorner off_2 \cdot \text{Off12}))) \\
\text{On12} &\stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5(r_1 + r_2)) \\
&\quad \curvearrowright \sigma^*_{\text{rel}}(\psi :\longrightarrow (\theta \ulcorner (off_1 \cdot \text{On2} + off_2 \cdot \text{On1})))
\end{aligned}$$

Heater example in ACP^{srt}_{hs}

- system with temperature limit
- $\theta \equiv (T_B^{\bullet} = {}^{\bullet}T_B) \quad \psi \equiv (T_B = 25)$

$$\mathrm{Start} \stackrel{def}{=} (T_B = T_0) \mathrel{\wedge\!\!\blacktriangle} \mathrm{Off12}$$

$$\mathrm{Off12} \stackrel{def}{=} (\dot{T}_B = -T_B) \curvearrowright\!\!\blacktriangledown\ \sigma^*_{\mathrm{rel}}(\theta \ulcorner\!\!\blacktriangledown (on_1 \cdot \mathrm{On1} + on_2 \cdot \mathrm{On2}))$$

$$\mathrm{On1} \stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5 r_1)$$
$$\curvearrowright\!\!\blacktriangledown\ \sigma^*_{\mathrm{rel}}((\theta \ulcorner\!\!\blacktriangledown on_2 \cdot \mathrm{On12}) + (\psi :\longrightarrow (\theta \ulcorner\!\!\blacktriangledown off_1 \cdot \mathrm{Off12})))$$

$$\mathrm{On2} \stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5 r_2)$$
$$\curvearrowright\!\!\blacktriangledown\ \sigma^*_{\mathrm{rel}}((\theta \ulcorner\!\!\blacktriangledown on_1 \cdot \mathrm{On12}) + (\psi :\longrightarrow (\theta \ulcorner\!\!\blacktriangledown off_2 \cdot \mathrm{Off12})))$$

$$\mathrm{On12} \stackrel{def}{=} (T_B \leq 25 \wedge \dot{T}_B = -T_B + 0.5(r_1 + r_2))$$
$$\curvearrowright\!\!\blacktriangledown\ \sigma^*_{\mathrm{rel}}(\psi :\longrightarrow (\theta \ulcorner\!\!\blacktriangledown (off_1 \cdot \mathrm{On2} + off_2 \cdot \mathrm{On1})))$$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$
  - associated ODEs: $\dot{\mathbf{X}} = \mathit{flow}(v)$
  - initial conditions: $\mathit{init}(v)$
  - invariants: $\mathit{inv}(v)$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$
  - associated ODEs: $\dot{\mathbf{X}} = \mathit{flow}(v)$
  - initial conditions: $\mathit{init}(v)$
  - invariants: $\mathit{inv}(v)$
- (control) switches: $e \in E$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$
  - associated ODEs: $\dot{\mathbf{X}} = \mathit{flow}(v)$
  - initial conditions: $\mathit{init}(v)$
  - invariants: $\mathit{inv}(v)$
- (control) switches: $e \in E$
  - events: $\mathit{event}(e) \in \mathcal{E}$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$
  - associated ODEs: $\dot{\mathbf{X}} = \mathit{flow}(v)$
  - initial conditions: $\mathit{init}(v)$
  - invariants: $\mathit{inv}(v)$
- (control) switches: $e \in E$
  - events: $\mathit{event}(e) \in \mathcal{E}$
  - predicate on $\mathbf{X}$: $\mathit{jump}(e)$
  - predicate on $\mathbf{X} \cup \mathbf{X}'$: $\mathit{reset}(e)$

Hybrid automata

- $(V, E, \mathbf{X}, \mathcal{E}, \mathit{flow}, \mathit{init}, \mathit{inv}, \mathit{event}, \mathit{jump}, \mathit{reset}, \mathit{urgent})$
- $\mathbf{X} = \{X_1, \ldots, X_n\}$, $\dot{X}_j$, X'_j
- control graph: $G = (V, E)$
- (control) modes: $v \in V$
  - associated ODEs: $\dot{\mathbf{X}} = \mathit{flow}(v)$
  - initial conditions: $\mathit{init}(v)$
  - invariants: $\mathit{inv}(v)$
- (control) switches: $e \in E$
  - events: $\mathit{event}(e) \in \mathcal{E}$
  - predicate on $\mathbf{X}$: $\mathit{jump}(e)$
  - predicate on $\mathbf{X} \cup \mathbf{X}'$: $\mathit{reset}(e)$
  - boolean: $\mathit{urgent}(e)$

HYPE model to hybrid automaton

- modes V: set of reachable configurations

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$
- if $v_j = \langle P_j, \sigma_j \rangle$ then

  $$\mathit{flow}(v_j)[X_i] \;=\; \sum\{r[\![I(\vec{W})]\!] \mid iv(\iota) = X_i \text{ and } \sigma_j(\iota) = (r, I(\vec{W}))\}$$

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$
- if $v_j = \langle P_j, \sigma_j \rangle$ then

  $$flow(v_j)[X_i] \;=\; \sum\{r[\![I(\vec{W})]\!] \mid iv(\iota) = X_i \text{ and } \sigma_j(\iota) = (r, I(\vec{W}))\}$$
- $inv(v) = true$

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$
- if $v_j = \langle P_j, \sigma_j \rangle$ then

  $$flow(v_j)[X_i] \;=\; \sum\{r[\![I(\vec{W})]\!] \mid iv(\iota) = X_i \text{ and } \sigma_j(\iota) = (r, I(\vec{W}))\}$$
- $inv(v) = true$
- let e be an edge associated with $\underline{\mathrm{a}}$ and let $ec(\underline{\mathrm{a}}) = (act_{\underline{\mathrm{a}}}, res_{\underline{\mathrm{a}}})$

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$
- if $v_j = \langle P_j, \sigma_j \rangle$ then

$$\mathit{flow}(v_j)[X_i] \;=\; \sum\{r[\![I(\vec{W})]\!] \mid iv(\iota) = X_i \text{ and } \sigma_j(\iota) = (r, I(\vec{W}))\}$$

- $\mathit{inv}(v) = \mathit{true}$
- let e be an edge associated with $\underline{a}$ and let $ec(\underline{a}) = (act_{\underline{a}}, res_{\underline{a}})$
  - $\mathit{event}(e) = \underline{a}$ and $\mathit{reset}(e) = res_{\underline{a}}$
  - if $act_{\underline{a}} \neq \perp$ then $\mathit{jump}(e) = act_{\underline{a}}$ and $\mathit{urgent}(e) = \mathit{true}$
    else $\mathit{jump}(e) = \mathit{true}$ and $\mathit{urgent}(e) = \mathit{false}$

HYPE model to hybrid automaton

- modes V: set of reachable configurations
- edges E: transitions between configurations
- variables $\mathbf{X}$: variables $\mathcal{V}$
- if $v_j = \langle P_j, \sigma_j \rangle$ then

$$\mathit{flow}(v_j)[X_i] \;=\; \sum\{r[\![I(\vec{W})]\!] \mid iv(\iota) = X_i \text{ and } \sigma_j(\iota) = (r, I(\vec{W}))\}$$

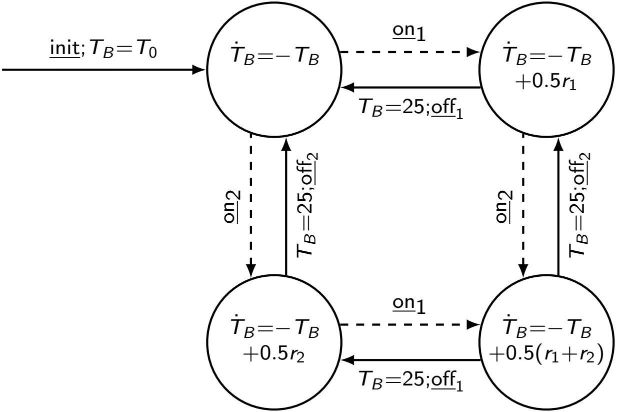
- $\mathit{inv}(v) = \mathit{true}$
- let e be an edge associated with $\underline{\mathrm{a}}$ and let $ec(\underline{\mathrm{a}}) = (\mathit{act}_{\underline{\mathrm{a}}}, \mathit{res}_{\underline{\mathrm{a}}})$
  - $\mathit{event}(e) = \underline{\mathrm{a}}$ and $\mathit{reset}(e) = \mathit{res}_{\underline{\mathrm{a}}}$
  - if $\mathit{act}_{\underline{\mathrm{a}}} \neq \perp$ then $\mathit{jump}(e) = \mathit{act}_{\underline{\mathrm{a}}}$ and $\mathit{urgent}(e) = \mathit{true}$
    else $\mathit{jump}(e) = \mathit{true}$ and $\mathit{urgent}(e) = \mathit{false}$
- $\mathit{init}(v) = \begin{cases} \mathit{res}_{\underline{\mathrm{init}}} & \text{if } v = \langle P, \sigma \rangle \text{ with primes removed} \\ \mathit{false} & \mathit{otherwise} \end{cases}$

Heater example (cont.)

- translation from HYPE system to hybrid automaton

Heater example (cont.)

- translation from HYPE system to hybrid automaton



Equivalence semantics

- system bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.

Equivalence semantics

- system bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- system bisimilar: $P \sim_s Q$ if in a system bisimulation

Equivalence semantics

- system bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- system bisimilar: $P \sim_s Q$ if in a system bisimulation
- *Theorem 1*: $\sim_s$ is a congruence for all operators

Equivalence semantics

- system bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- system bisimilar: $P \sim_s Q$ if in a system bisimulation
- *Theorem 1*: $\sim_s$ is a congruence for all operators
- *Theorem 2*: if Σ_1 and Σ_2 have the same prefixes then $\Sigma_1 \underset{L}{\bowtie} \underline{\text{init}}.Con \sim_s \Sigma_2 \underset{L}{\bowtie} \underline{\text{init}}.Con$, assuming well-defined systems

Equivalence semantics

- system bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- system bisimilar: $P \sim_s Q$ if in a system bisimulation
- *Theorem 1*: $\sim_s$ is a congruence for all operators
- *Theorem 2*: if Σ_1 and Σ_2 have the same prefixes then $\Sigma_1 \underset{L}{\bowtie} \underline{\text{init}}.Con \sim_s \Sigma_2 \underset{L}{\bowtie} \underline{\text{init}}.Con$, assuming well-defined systems
- *Theorem 3*: if $P \sim_s Q$ then $P_\sigma = Q_\sigma$ for all σ, assuming well-defined systems

Heater example (cont.)

- Consider two fans in Room C and none in Room A

$$Sys' \stackrel{def}{=} (Fan_{1,C,B} \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Room_B(T_B)$$

$$MF' \stackrel{def}{=} Sys' \underset{M}{\bowtie} \underline{\mathrm{init}}.Con \qquad M = \{\underline{\mathrm{init}}, \underline{\mathrm{on}}_1, \underline{\mathrm{off}}_1, \underline{\mathrm{on}}_2, \underline{\mathrm{off}}_2\}$$

Heater example (cont.)

- Consider two fans in Room C and none in Room A

$$Sys' \stackrel{def}{=} (Fan_{1,C,B} \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Room_B(T_B)$$

$$MF' \stackrel{def}{=} Sys' \underset{M}{\bowtie} \underline{\mathrm{init}}.Con \qquad M = \{\underline{\mathrm{init}}, \underline{\mathrm{on}}_1, \underline{\mathrm{off}}_1, \underline{\mathrm{on}}_2, \underline{\mathrm{off}}_2\}$$

Heater example (cont.)

- Consider two fans in Room C and none in Room A

$$Sys' \stackrel{def}{=} (Fan_{1,C,B} \underset{\{\underline{init}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{init}\}}{\bowtie} Room_B(T_B)$$

$$MF' \stackrel{def}{=} Sys' \underset{M}{\bowtie} \underline{init}.Con \qquad M = \{\underline{init}, \underline{on}_1, \underline{off}_1, \underline{on}_2, \underline{off}_2\}$$

- Sys and Sys' have the same prefixes

$$\bigcup_{i=1,2} \{\underline{init} : (t_{i,y}, 0, const), \underline{on}_i : (t_{i,y}, r_i, const_{adj}), \underline{off}_i : (t_{i,y}, 0, const)\}$$

$$\cup \ \{\underline{init} : (t_{0,x}, -1, linear(T))\}$$

Heater example (cont.)

- Consider two fans in Room C and none in Room A

$$Sys' \stackrel{def}{=} (Fan_{1,C,B} \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\mathrm{init}}\}}{\bowtie} Room_B(T_B)$$

$$MF' \stackrel{def}{=} Sys' \underset{M}{\bowtie} \underline{\mathrm{init}}.Con \qquad M = \{\underline{\mathrm{init}}, \underline{\mathrm{on}}_1, \underline{\mathrm{off}}_1, \underline{\mathrm{on}}_2, \underline{\mathrm{off}}_2\}$$

- Sys and Sys' have the same prefixes

$$\bigcup_{i=1,2} \{\underline{\mathrm{init}}:(t_{i,y}, 0, const), \underline{\mathrm{on}}_i:(t_{i,y}, r_i, const_{adj}), \underline{\mathrm{off}}_i:(t_{i,y}, 0, const)\}$$

$$\cup\ \{\underline{\mathrm{init}}:(t_{0,x}, -1, linear(T))\}$$

- by Theorem 2, $MF \sim_s MF'$

Heater example (cont.)

- Consider two fans in Room C and none in Room A

$$Sys' \stackrel{def}{=} (Fan_{1,C,B} \underset{\{\underline{\text{init}}\}}{\bowtie} Fan_{2,C,B}) \underset{\{\underline{\text{init}}\}}{\bowtie} Room_B(T_B)$$

$$MF' \stackrel{def}{=} Sys' \underset{M}{\bowtie} \underline{\text{init}}.Con \qquad M = \{\underline{\text{init}}, \underline{\text{on}}_1, \underline{\text{off}}_1, \underline{\text{on}}_2, \underline{\text{off}}_2\}$$

- Sys and Sys' have the same prefixes

$$\bigcup_{i=1,2} \{\underline{\text{init}}:(t_{i,y}, 0, const), \underline{\text{on}}_i:(t_{i,y}, r_i, const_{adj}), \underline{\text{off}}_i:(t_{i,y}, 0, const)\}$$

$$\cup\ \{\underline{\text{init}}:(t_{0,x}, -1, linear(T))\}$$

- by Theorem 2, $MF \sim_s MF'$
- by Theorem 3, MF and MF' have the same ODEs

Bisimulations for ACP^{srt}_{hs}

- bisimulation: relation B if for all $(\langle P, \sigma\rangle, \langle Q, \sigma\rangle) \in B$ whenever
  1. $\langle P, \sigma\rangle \xrightarrow{\text{a}} \langle P', \sigma'\rangle$, there exists $\langle Q', \sigma'\rangle$ with $\langle Q, \sigma\rangle \xrightarrow{\text{a}} \langle Q', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.
  2. $\langle Q, \sigma\rangle \xrightarrow{\text{a}} \langle Q', \sigma'\rangle$, there exists $\langle P', \sigma'\rangle$ with $\langle P, \sigma\rangle \xrightarrow{\text{a}} \langle P', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.

Bisimulations for ACP^{srt}_{hs}

- bisimulation: relation B if for all $(\langle P, \sigma \rangle, \langle Q, \sigma \rangle) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(\langle P', \sigma' \rangle, \langle Q', \sigma' \rangle) \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(\langle P', \sigma' \rangle, \langle Q', \sigma' \rangle) \in B$.
- bisimilar: $\langle P, \sigma \rangle \leftrightarrows \langle Q, \sigma \rangle$ if in a bisimulation

Bisimulations for ACP^{srt}_{hs}

- bisimulation: relation B if for all $(\langle P, \sigma\rangle, \langle Q, \sigma\rangle) \in B$ whenever
  1. $\langle P, \sigma\rangle \xrightarrow{\underline{a}} \langle P', \sigma'\rangle$, there exists $\langle Q', \sigma'\rangle$ with $\langle Q, \sigma\rangle \xrightarrow{\underline{a}} \langle Q', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.
  2. $\langle Q, \sigma\rangle \xrightarrow{\underline{a}} \langle Q', \sigma'\rangle$, there exists $\langle P', \sigma'\rangle$ with $\langle P, \sigma\rangle \xrightarrow{\underline{a}} \langle P', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.
- bisimilar: $\langle P, \sigma\rangle \leftrightarroweq \langle Q, \sigma\rangle$ if in a bisimulation
- bisimilar: $P \leftrightarroweq Q$ if $\langle P, \sigma\rangle \leftrightarroweq \langle Q, \sigma\rangle$ for all σ

Bisimulations for ACP^{srt}_{hs}

- bisimulation: relation B if for all $(\langle P, \sigma\rangle, \langle Q, \sigma\rangle) \in B$ whenever
  1. $\langle P, \sigma\rangle \xrightarrow{a} \langle P', \sigma'\rangle$, there exists $\langle Q', \sigma'\rangle$ with $\langle Q, \sigma\rangle \xrightarrow{a} \langle Q', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.
  2. $\langle Q, \sigma\rangle \xrightarrow{a} \langle Q', \sigma'\rangle$, there exists $\langle P', \sigma'\rangle$ with $\langle P, \sigma\rangle \xrightarrow{a} \langle P', \sigma'\rangle$ and $(\langle P', \sigma'\rangle, \langle Q', \sigma'\rangle) \in B$.
- bisimilar: $\langle P, \sigma\rangle \leftrightarroweq \langle Q, \sigma\rangle$ if in a bisimulation
- bisimilar: $P \leftrightarroweq Q$ if $\langle P, \sigma\rangle \leftrightarroweq \langle Q, \sigma\rangle$ for all σ
- not a congruence for parallel operator of ACP^{srt}_{hs}

Bisimulations for ACP_{hs}^{srt} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.

Bisimulations for ACP^{srt}_{hs} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- ic-bisimilar: $P \leftrightarrows Q$ if in a bisimulation

Bisimulations for ACP^{srt}_{hs} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- ic-bisimilar: $P \underline{\leftrightarrows} Q$ if in a bisimulation
- congruence for all ACP^{srt}_{hs} operators

Bisimulations for ACP^{srt}_{hs} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{\underline{a}} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\underline{a}} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- ic-bisimilar: $P \leftrightarrows\!\!\!\!\!= Q$ if in a bisimulation
- congruence for all ACP^{srt}_{hs} operators
- identical definition to system bisimulation

Bisimulations for ACP^{srt}_{hs} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{a} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{a} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{a} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{a} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- ic-bisimilar: $P \mathrel{\overset{\leftrightarrows}{=}} Q$ if in a bisimulation
- congruence for all ACP^{srt}_{hs} operators
- identical definition to system bisimulation
- *Theorem 4*: for well-defined HYPE models
  $$P \sim_s Q \;\Leftrightarrow\; P \mathrel{\overset{\leftrightarrows}{=}} Q \;\Leftrightarrow\; P \mathrel{\underline{\leftrightarrows}} Q$$

Bisimulations for ACP^{srt}_{hs} (cont.)

- ic-bisimulation: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{a} \langle P', \sigma' \rangle$, there exists $\langle Q', \sigma' \rangle$ with $\langle Q, \sigma \rangle \xrightarrow{a} \langle Q', \sigma' \rangle$ and $(P', Q') \in B$.
  2. $\langle Q, \sigma \rangle \xrightarrow{a} \langle Q', \sigma' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{a} \langle P', \sigma' \rangle$ and $(P', Q') \in B$.
- ic-bisimilar: $P \overset{\leftrightarrows}{=} Q$ if in a bisimulation
- congruence for all ACP^{srt}_{hs} operators
- identical definition to system bisimulation
- *Theorem 4*: for well-defined HYPE models
  $$P \sim_s Q \;\Leftrightarrow\; P \overset{\leftrightarrows}{=} Q \;\Leftrightarrow\; P \leftrightarrows Q$$
  - $\langle P, \sigma \rangle \xrightarrow{a} \langle P', \sigma[u_1, \ldots, u_n] \rangle$, $\langle Q, \sigma \rangle \xrightarrow{a} \langle Q', \sigma[u_1, \ldots, u_n] \rangle$

More general bisimulation

- let $\equiv$ be an equivalence over states
- system bisimulation with respect to $\equiv$: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \tau' \rangle$ with $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  2. $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  3. $\sigma \equiv \tau$

More general bisimulation

- let $\equiv$ be an equivalence over states
- system bisimulation with respect to $\equiv$: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \tau' \rangle$ with $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  2. $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  3. $\sigma \equiv \tau$

More general bisimulation

- let $\equiv$ be an equivalence over states
- system bisimulation with respect to $\equiv$: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma\rangle \xrightarrow{\text{a}} \langle P', \sigma'\rangle$, there exists $\langle Q', \tau'\rangle$ with $\langle Q, \tau\rangle \xrightarrow{\text{a}} \langle Q', \tau'\rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  2. $\langle Q, \tau\rangle \xrightarrow{\text{a}} \langle Q', \tau'\rangle$, there exists $\langle P', \sigma'\rangle$ with $\langle P, \sigma\rangle \xrightarrow{\text{a}} \langle P', \sigma'\rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  3. $\sigma \equiv \tau$
- system bisimilar with respect to $\equiv$: $P \sim_s^{\equiv} Q$ if in a system bisimulation with respect to $\equiv$

More general bisimulation

- let $\equiv$ be an equivalence over states
- system bisimulation with respect to $\equiv$: relation B if for all $(P, Q) \in B$ whenever
  1. $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, there exists $\langle Q', \tau' \rangle$ with $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  2. $\langle Q, \tau \rangle \xrightarrow{\mathsf{a}} \langle Q', \tau' \rangle$, there exists $\langle P', \sigma' \rangle$ with $\langle P, \sigma \rangle \xrightarrow{\mathsf{a}} \langle P', \sigma' \rangle$, $\sigma' \equiv \tau'$ and $(P', Q') \in B$.
  3. $\sigma \equiv \tau$
- system bisimilar with respect to $\equiv$: $P \sim_s^{\equiv} Q$ if in a system bisimulation with respect to $\equiv$
- system bisimulation is $\sim_s^{=}$

More general bisimulation (cont.)

- $\equiv$ preserves updating if

$$\sigma \equiv \tau \Rightarrow \sigma[\iota \mapsto (r, I)] \equiv \tau[\iota \mapsto (r, I)]$$

More general bisimulation (cont.)

- $\equiv$ preserves updating if

$$\sigma \equiv \tau \Rightarrow \sigma[\iota \mapsto (r, I)] \equiv \tau[\iota \mapsto (r, I)]$$

- *Theorem 5*: if $\equiv$ preserves updating then $\sim_s^{\equiv}$ is a congruence for all operators

More general bisimulation (cont.)

- $\equiv$ preserves updating if

$$\sigma \equiv \tau \Rightarrow \sigma[\iota \mapsto (r, I)] \equiv \tau[\iota \mapsto (r, I)]$$

- *Theorem 5*: if $\equiv$ preserves updating then $\sim_s^{\equiv}$ is a congruence for all operators
- what are interesting equivalences over states?

$$P \sim_s^{\equiv} Q \quad \stackrel{?}{\Rightarrow} \quad P_\sigma = Q_\sigma$$

More general bisimulation (cont.)

- $\sigma_1 \doteq \sigma_2$ if

  $$add(\sigma_1, V, f(\vec{X})) = add(\sigma_2, V, f(\vec{X})) \text{ for all } V,\ f(\vec{X}) \text{ where}$$

  $$add(\sigma, V, f(\vec{X})) = \sum\{r \mid \mathrm{iv}(\iota) = V, \sigma(\iota) = (r, I(\vec{X})), f(\vec{X}) = [\![I(\vec{X})]\!]\}$$

- $P \sim_s^{\doteq} Q \quad \Rightarrow \quad P_\sigma = Q_\sigma$
- $\doteq$ does not preserve updating
- solutions
  - redefine *add* to preserve updates
  - require iv to be injective
  - consider individual equivalences

Conclusions and further work

- HYPE

Conclusions and further work

- HYPE
  - process algebra for hybrid systems

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence
- modelling: dual-tank system, bottling line, Repressilator

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence
- modelling: dual-tank system, bottling line, Repressilator
- further work

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence
- modelling: dual-tank system, bottling line, Repressilator
- further work
  - branching time versus linear time equivalences

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence
- modelling: dual-tank system, bottling line, Repressilator
- further work
  - branching time versus linear time equivalences
  - usefulness of equivalences

Conclusions and further work

- HYPE
  - process algebra for hybrid systems
  - describe flows, fine-grained modelling technique
  - operational semantics provide ODEs
  - map to hybrid automata
  - system bisimulation
  - relationship between bisimulation and ODEs
  - other bisimulations, more general equivalence
- modelling: dual-tank system, bottling line, Repressilator
- further work
  - branching time versus linear time equivalences
  - usefulness of equivalences
  - more modelling, systems biology

Thank you