A Formal Framework for the Analysis of Human-Machine Interactions

(Private Defense)

Sébastien Combéfis[1]

[1]Université catholique de Louvain (UCL)

ICT, Electronics and Applied Mathematics Institute (ICTEAM)

September 11, 2013

UCL Université catholique de Louvain

icteam

Motivation

State of the Art

- Systematic analysis of a well-known case study for mode confusion (Rushby)
- Using model checking to verify usability properties (Campos)
- Automatically generate adequate user interface (Degani)

State of the Art

- Systematic analysis of a well-known case study for mode confusion (Rushby)
- Using model checking to verify usability properties (Campos)
- Automatically generate adequate user interface (Degani)

Adequate interaction — Free of automation surprises

Research Goals

- Modelling the elements involved in human-machine interaction

  A formal model to ease the analyses of interaction

- Defining a property characterising safe interactions

  Interaction free of automation surprises

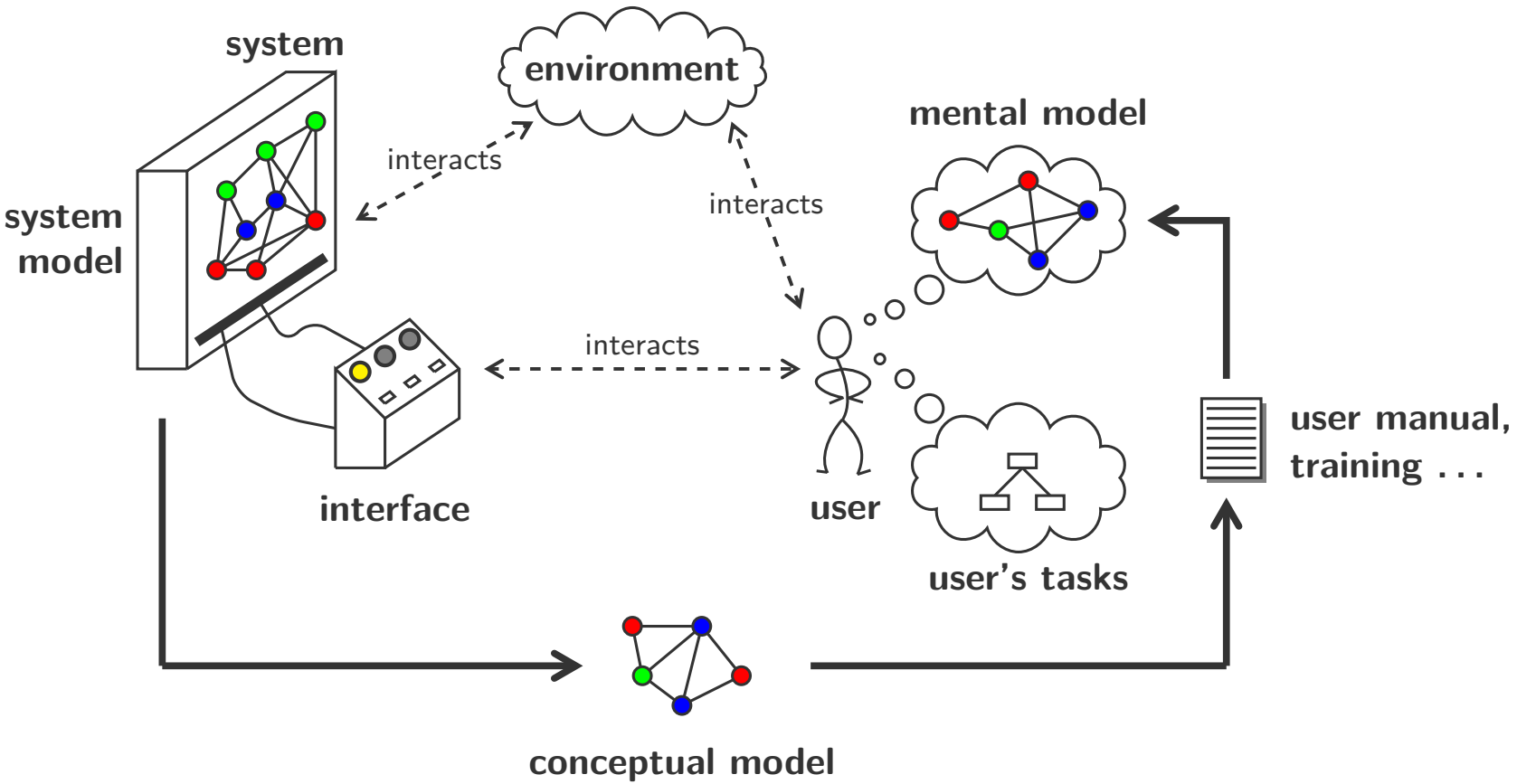
- Devising algorithms to support analyses of HMI

  Automatic conceptual model generation algorithm

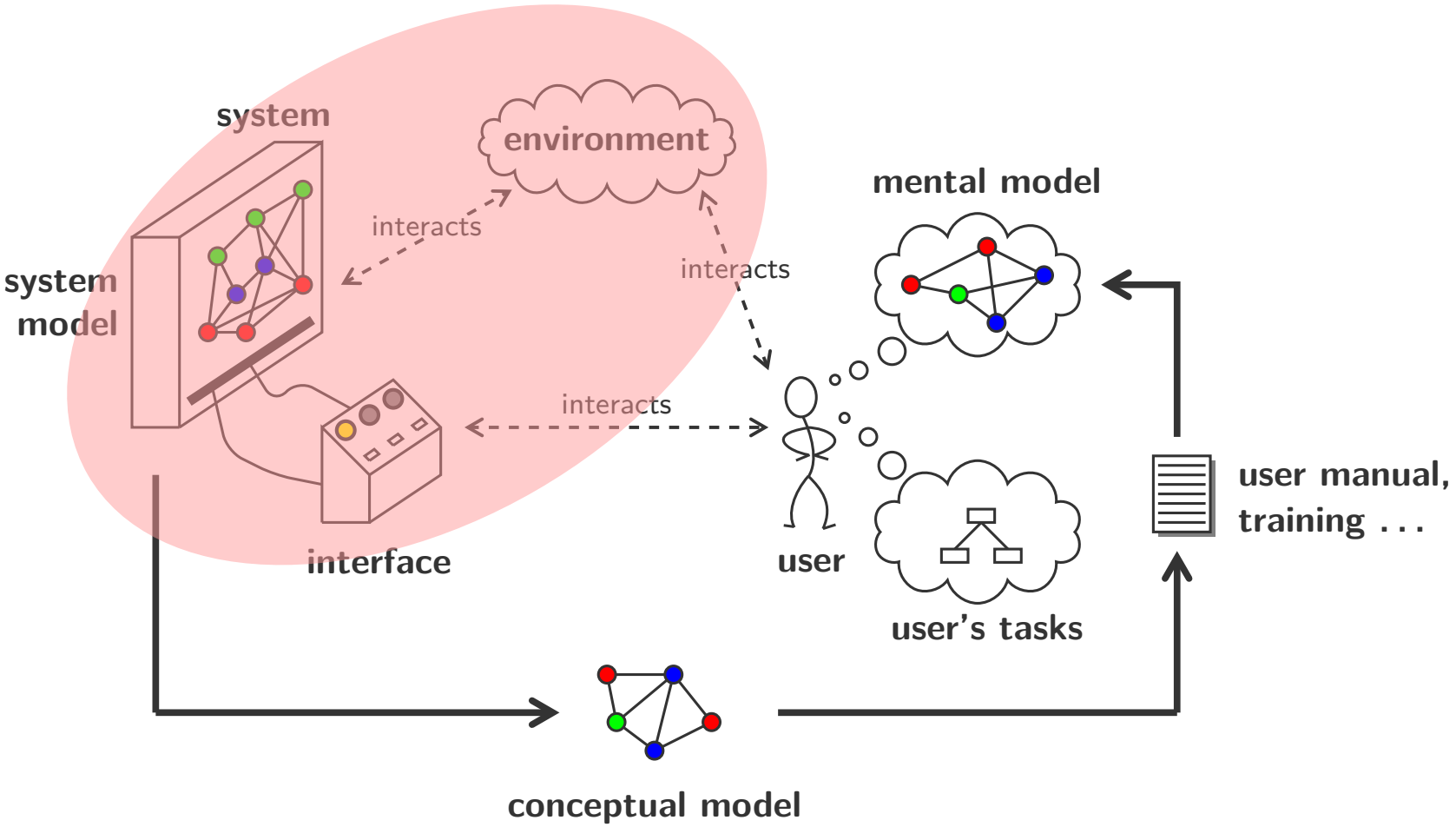
- Testing the proposed techniques on case studies

Outline

Human-Machine Interaction



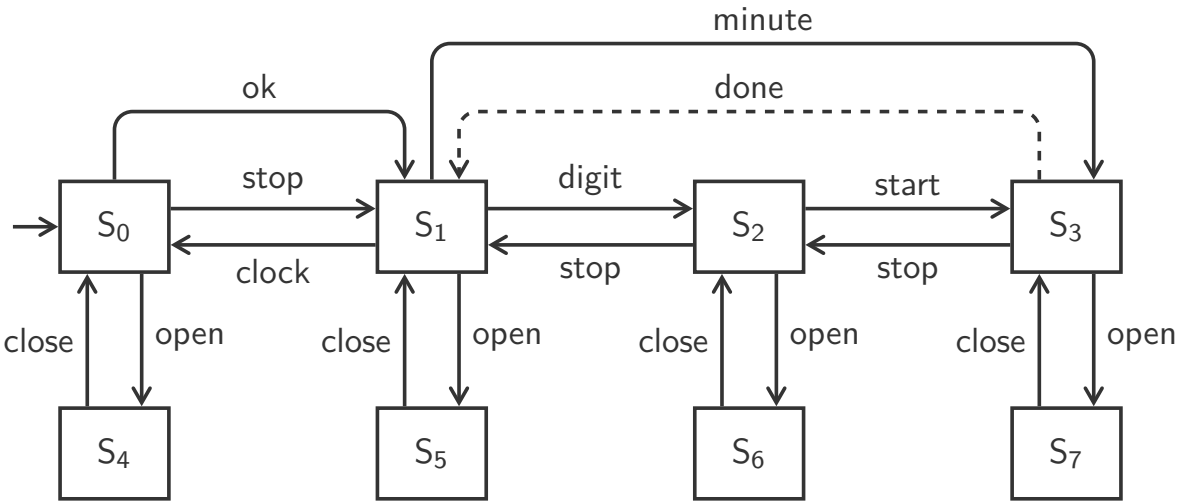
Human-Machine Interaction



Mental and conceptual models used to represent the same model

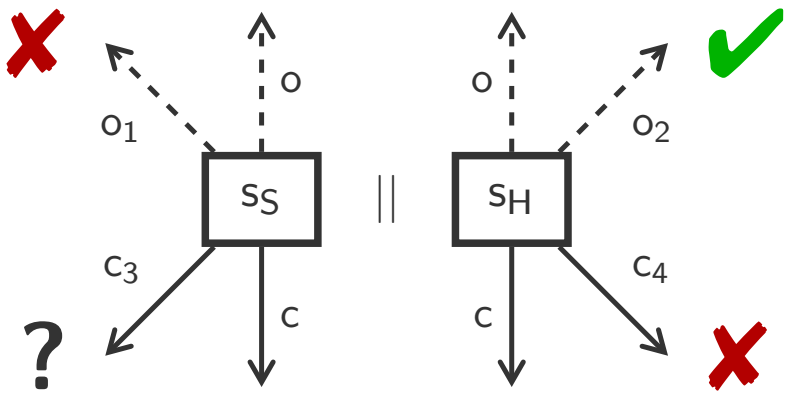
Formal Modelling

- HMI-LTS extends LTS with inputs and outputs:
  - Commands executed by the user
  - Observations executed by the system and observed by the user
  - Internal actions invisible to the user



Interaction Model

- **Interaction:**
  - Represented with the synchronous parallel composition



- **Bad situations:**
  - A command missing on the system model (c_4)
  - An observation missing on the mental model (o_1)

Full-control property

- Full-control property captures safe interaction
- During the interaction between a user and a system:
  - The user must know exactly the possible commands...
  - ...and at least all the possible observations

$\mathcal{H}$ fc $\mathcal{S}$ if and only if :

$\forall \sigma \in \mathcal{L}^*$ such that $s_S \in (s_{0_S}$ **after** $\sigma)$ and $s_H \in (s_{0_H}$ **after** $\sigma)$:

$$A^c(s_S) = A^c(s_H) \quad \text{and} \quad A^o(s_S) \subseteq A^o(s_H)$$

Full-control determinism

- Two states are fc-compatible if they exhibit the same behaviour for the operator

$s_1 \approx_{\text{fc}} s_2$ if and only if :

$\forall \sigma \in \mathcal{L}^*$ such that $s_1' \in (s_1 \textbf{ after } \sigma)$ and $s_2' \in (s_2 \textbf{ after } \sigma)$:

$A^c(s_1') = A^c(s_2')$

- A system model is fc-deterministic if: $s_0 \approx_{\text{fc}} s_0$

$\mathcal{S}$ is fc-deterministic $\implies det(\mathcal{S})\,\text{fc}\,\mathcal{S}$

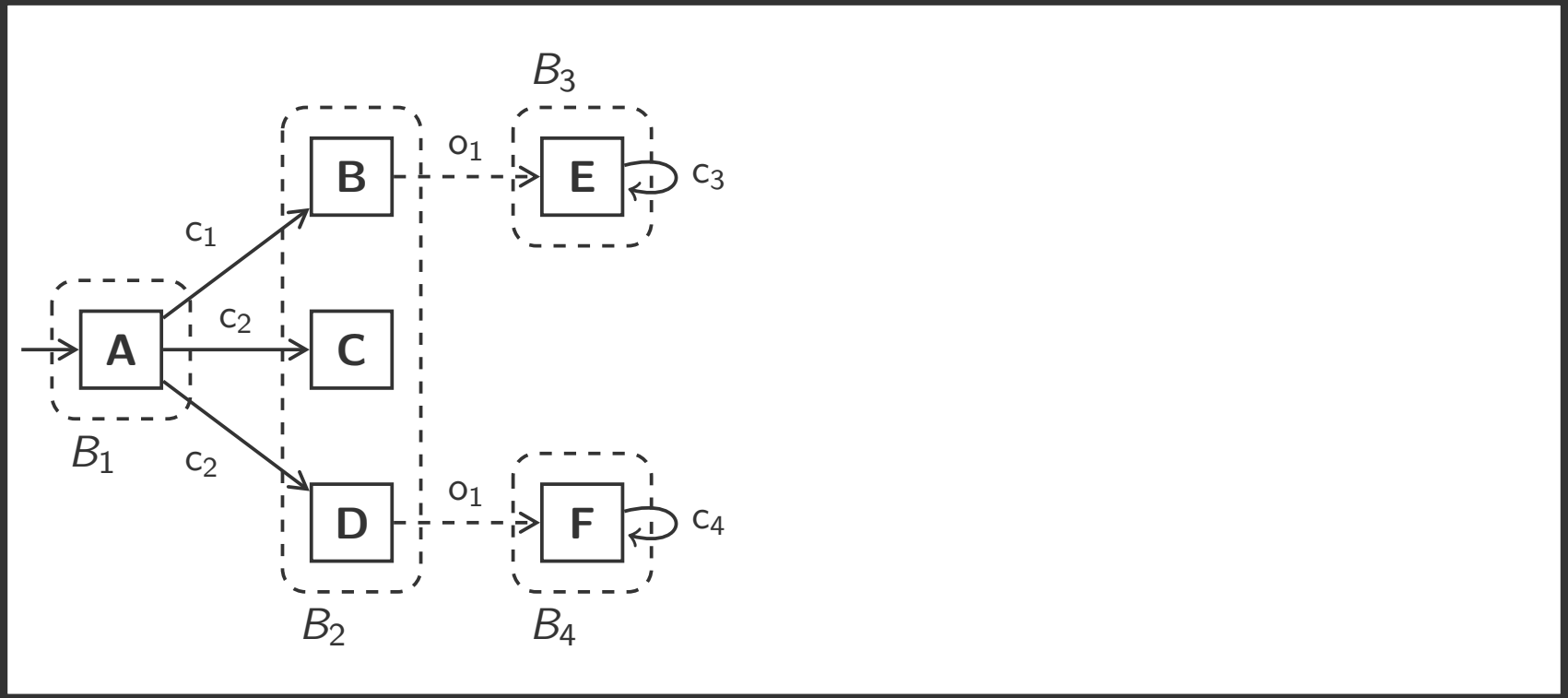
$det(\mathcal{S})$ is one possible full-control mental model

Generation Problem

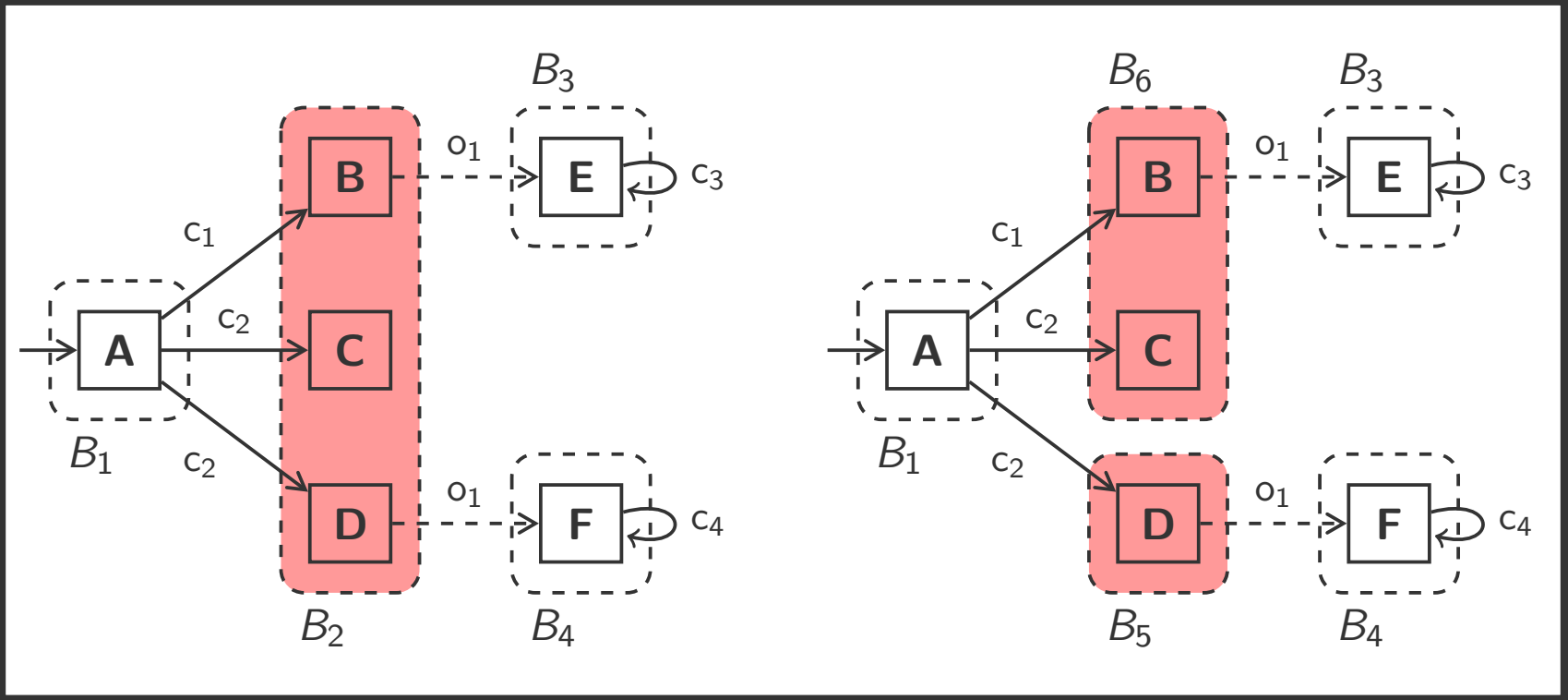
- **Goal:** Given the model of a system, automatically generate a minimal full-control conceptual model
- **Motivation:**
  - Extract the minimal behaviour of the system, so that it can be controlled without surprise
  - Help to build artifacts: manuals, procedures, trainings, ...
  - If such abstraction does not exist, provide feedback to help redesigning the system

Reduction-based algorithm

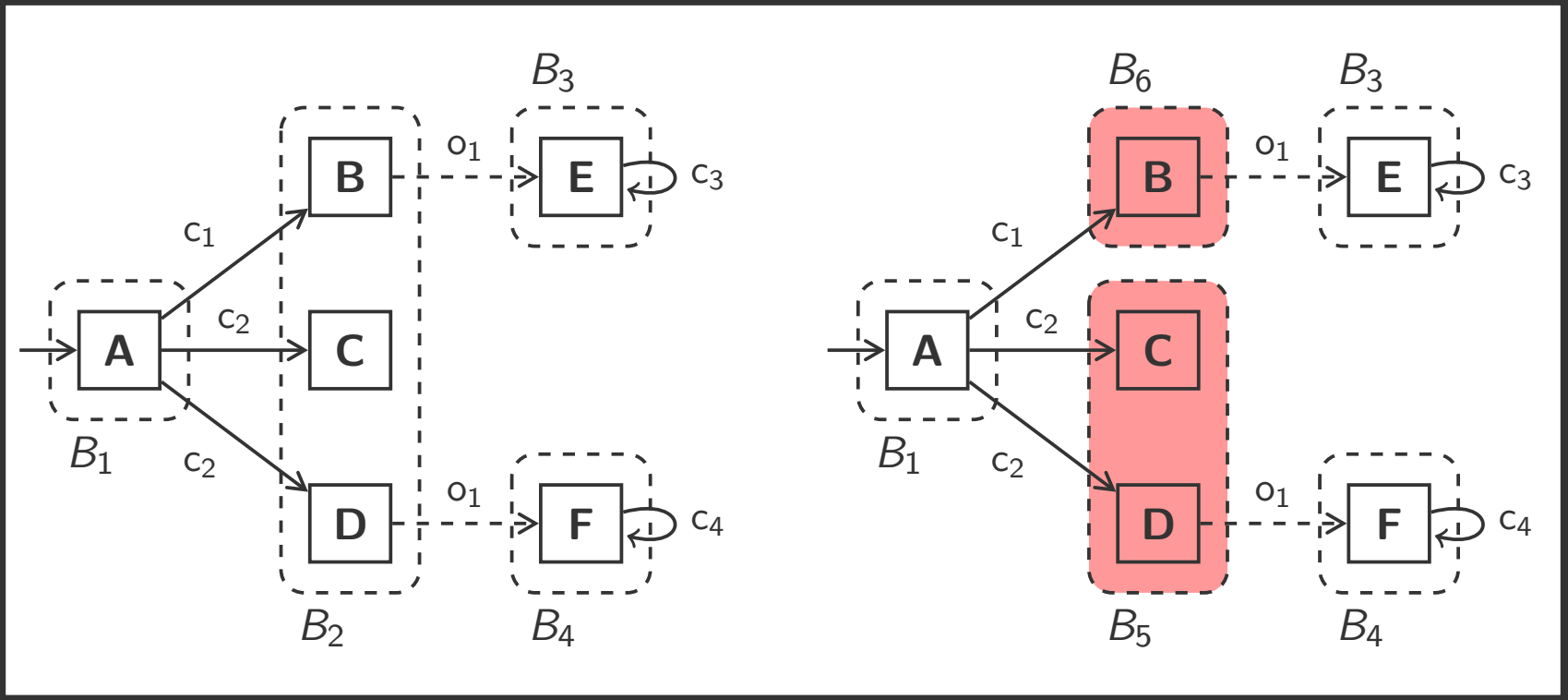
Reduction-based algorithm



Reduction-based algorithm



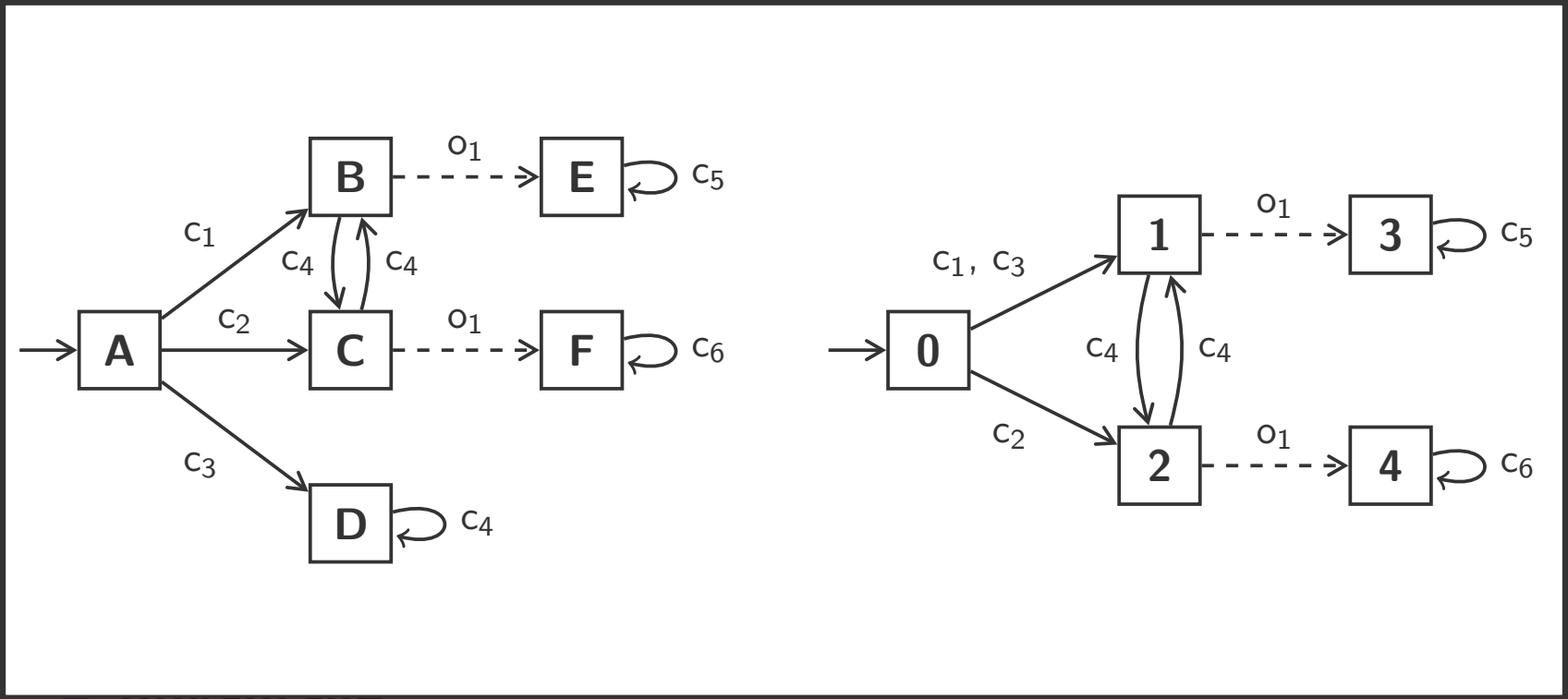
Reduction-based algorithm



Reduction-based algorithm

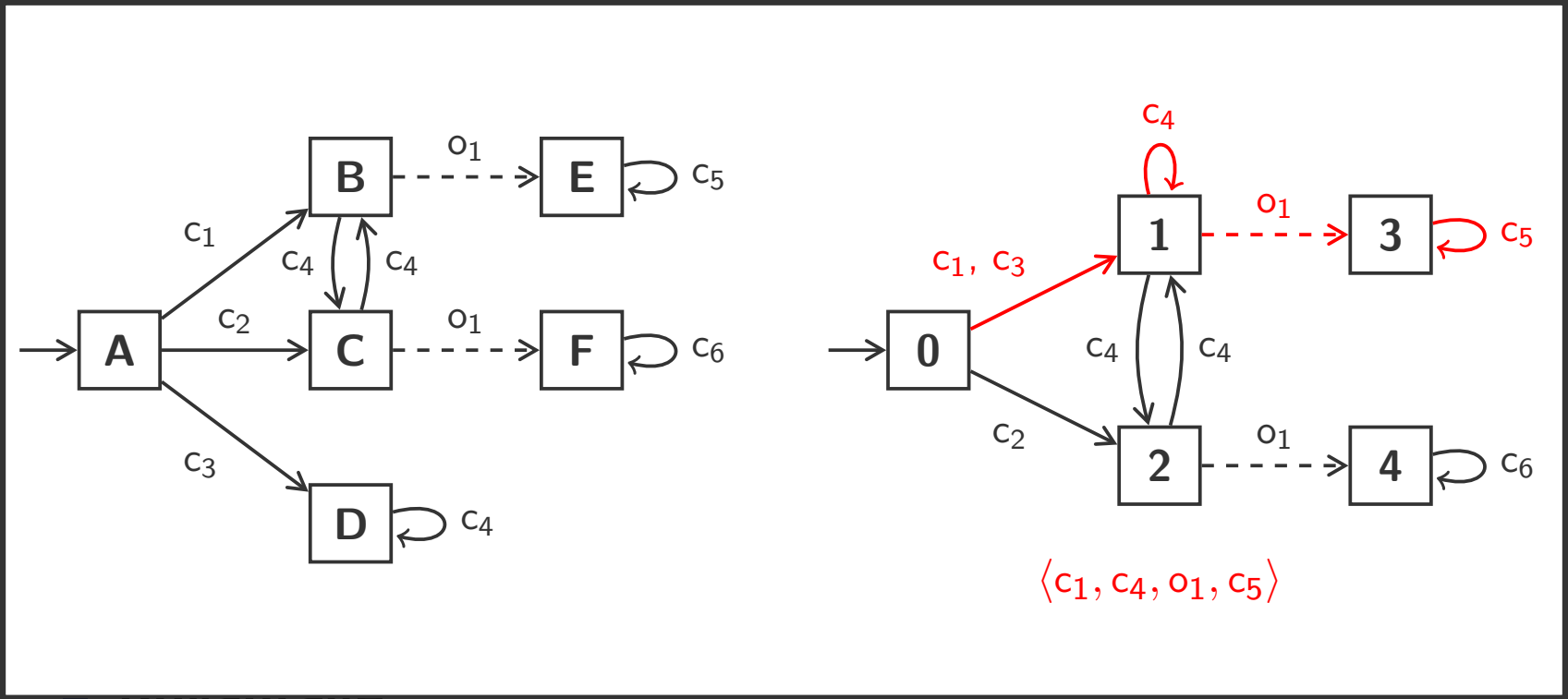
- **Drawback:**
  - Cannot always find the minimal full-control conceptual model
- **Advantage:**
  - Good time complexity (if not searching optimal solution)

Reduction-based algorithm



- Advantage:
  - Good time complexity (if not searching optimal solution)

Reduction-based algorithm



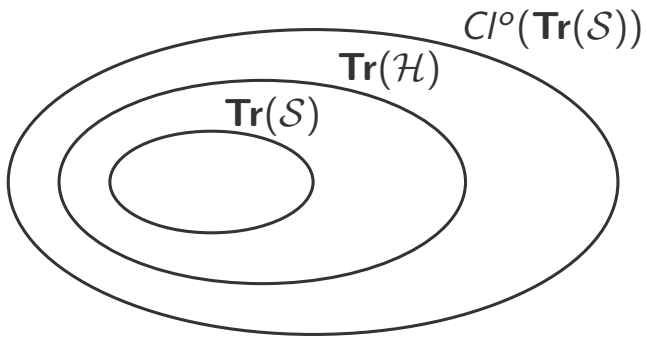
- Advantage:
  - Good time complexity (if not searching optimal solution)

3DFA characterisation

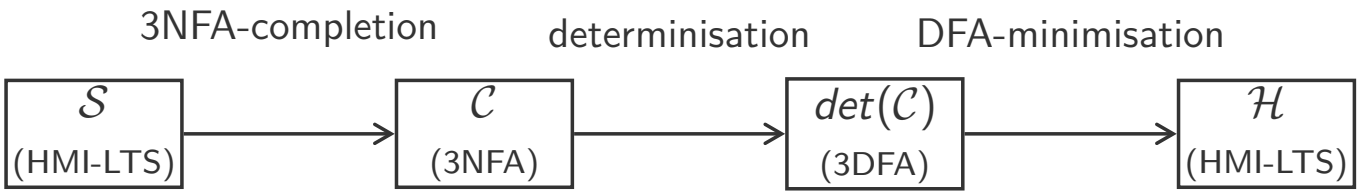
- The full-control property is characterised by the traces of the system

$$\mathcal{H} \,\mathrm{fc}\, \mathcal{S} \iff L_{Acc} \subseteq \mathbf{Tr}(\mathcal{H}) \subseteq \overline{L_{Rej}}$$

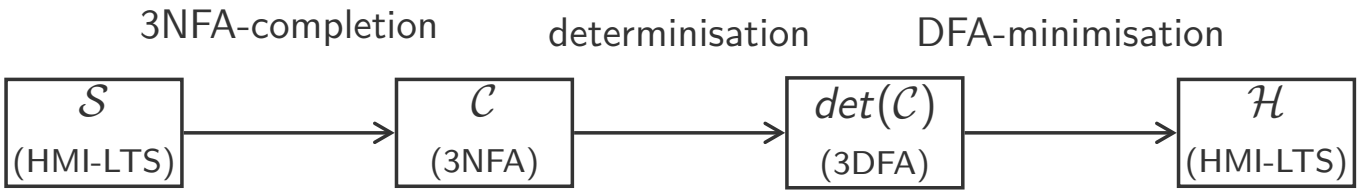
Where L_{Acc} and L_{Rej} are the languages accepted and rejected by the 3DFA characterising $\mathcal{S}$



3DFA-based algorithm

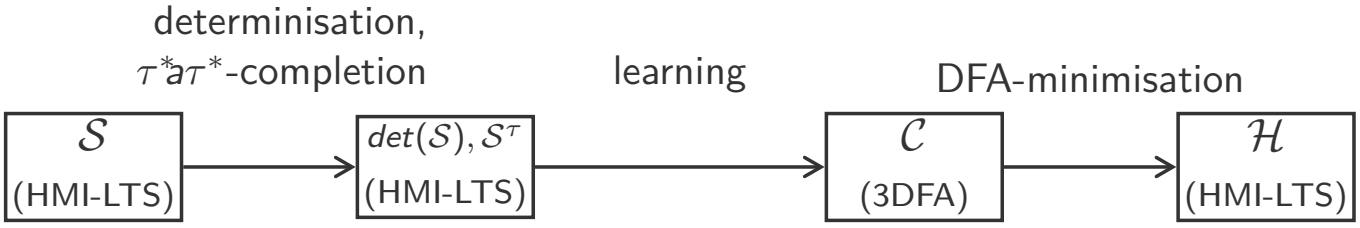


3DFA-based algorithm

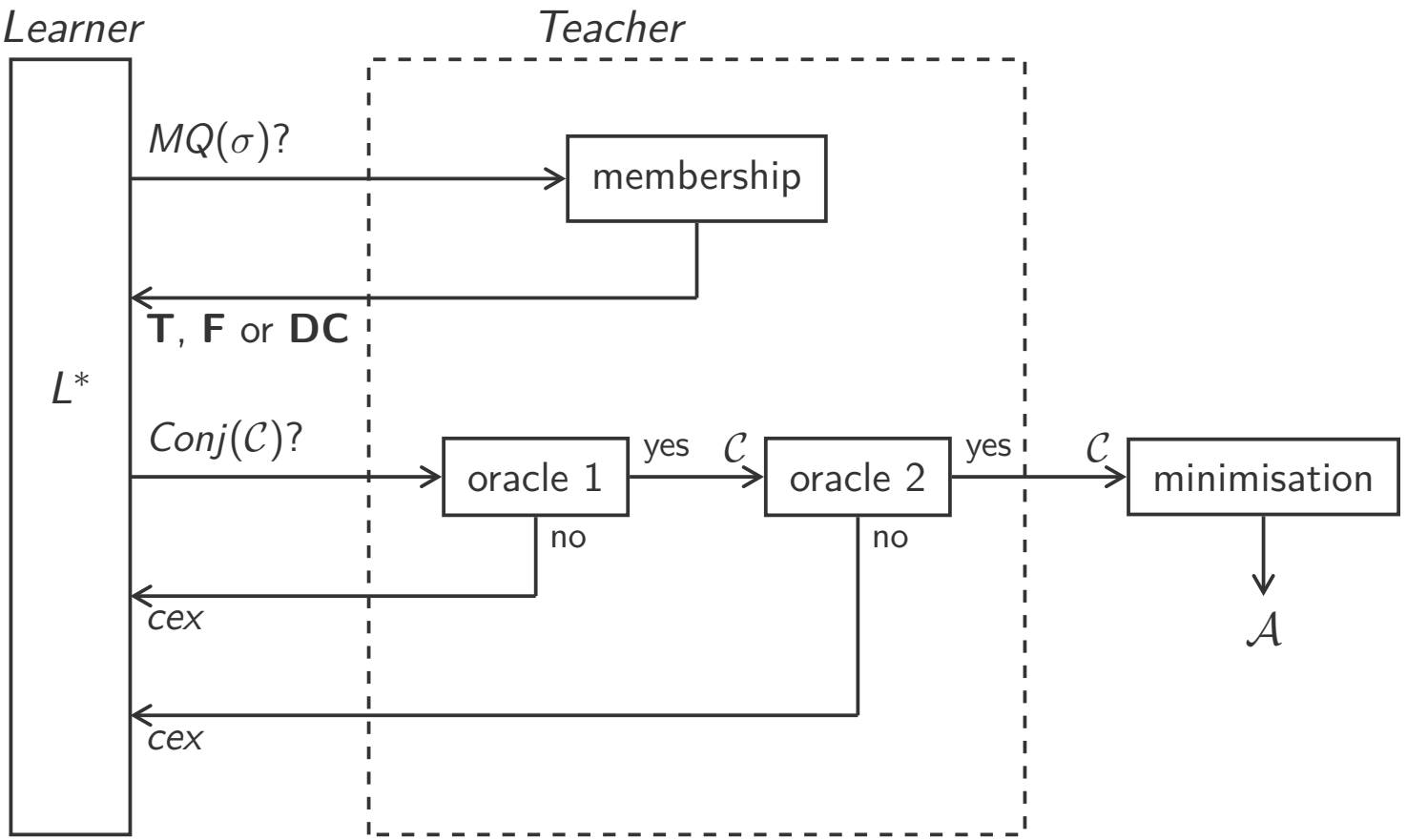


- **Drawback:**
  - Determinisation/minimisation performed on large automata (time and space explosion)
- **Advantage:**
  - Computes the minimal full-control conceptual model

Learning-based algorithm

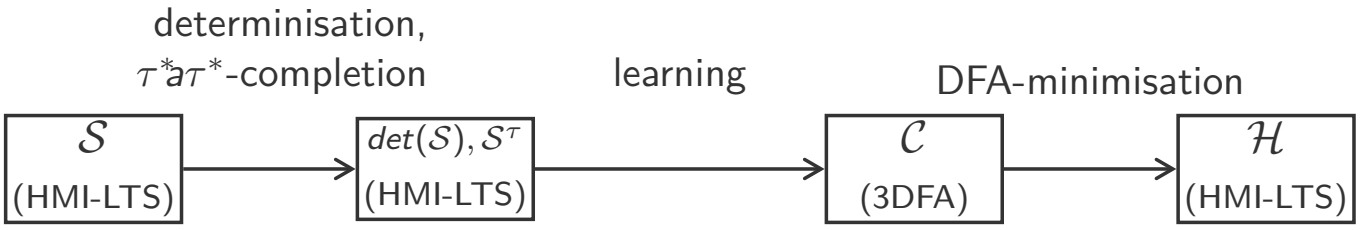


Learning-based algorithm



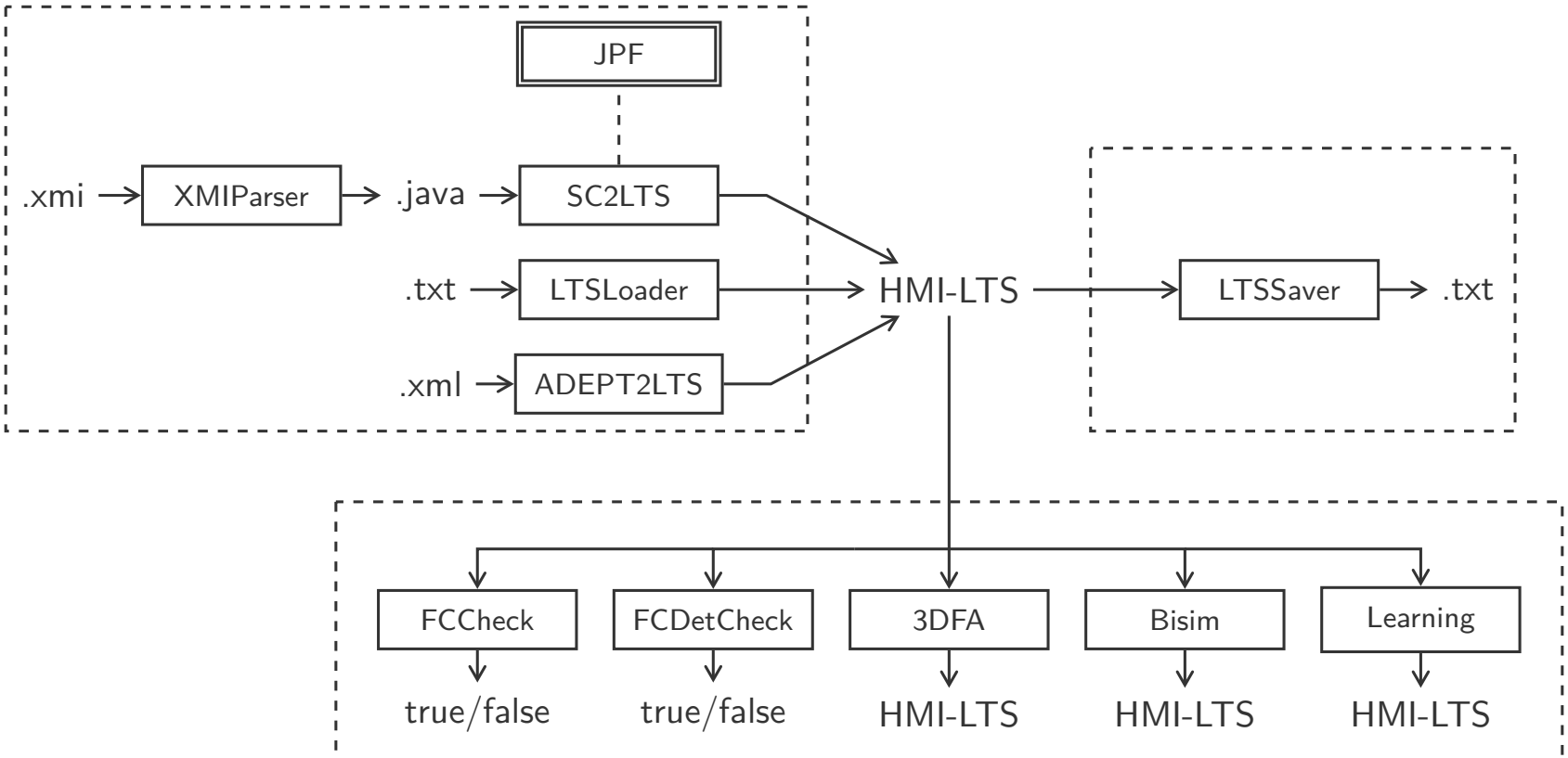
Model checking problems

Learning-based algorithm



- **Drawback:**
  - Queries to teacher can be time consuming
- **Advantage:**
  - Counterexample is short and directly provided

The jpf-hmi tool



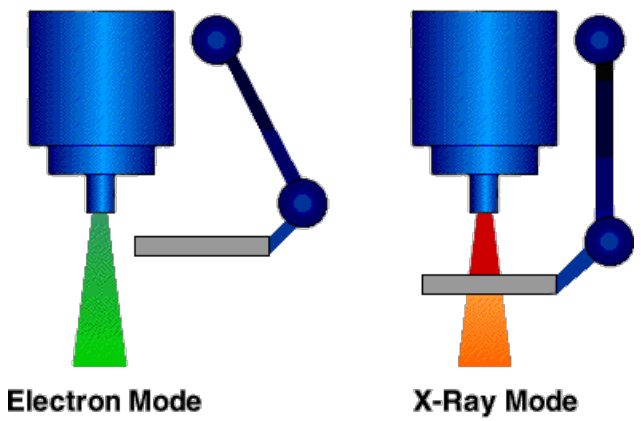
Performance

- Comparison of the three generation algorithms
  - No clear winner, depends on the instances
  - Different ways to handle non-fc-determinism

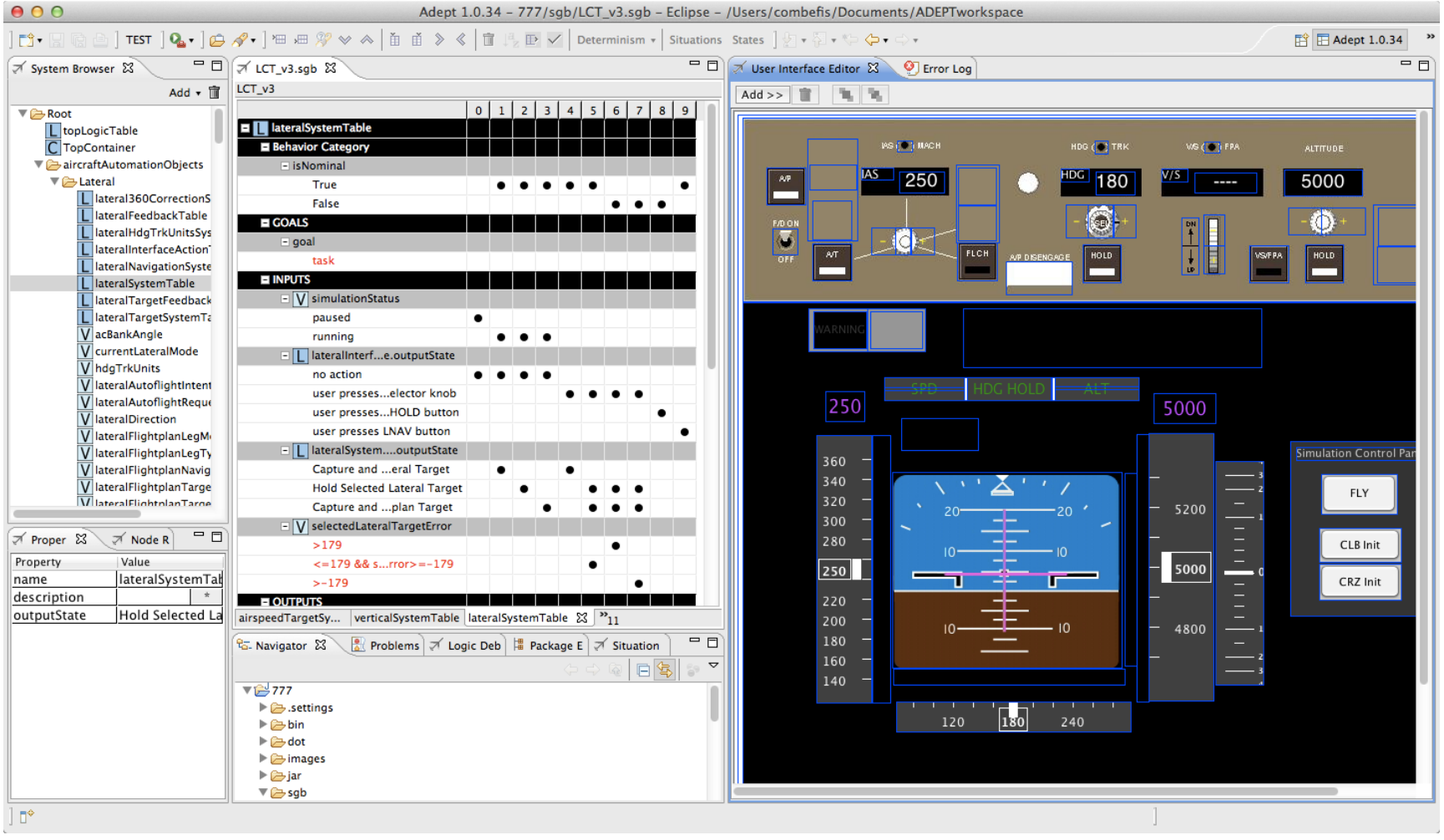
	System	Mental	3DFA	Reduction	Learning
VTS	8/20	5/14	26 ms (10)	25 ms	75 ms (10)
Therac-25	136/448	24/83	**63 ms (138)**	375 ms	1920 ms (70)
Air Cond.	13608/44748	222/834	–	2684 ms	–
TimedVCR	3352/15082	2/9	22595 ms (3354)	479 ms	**988 ms (6)**
Countdown-2	11/27	8/18	33 ms (10)	27 ms	155 ms (10)
Countdown-12	55/137	30/73	113 ms (32)	**54 ms**	3576 ms (32)

Applicability

- Analysis of realistic examples (Therac-25, microwave, VCR)
- Finding existing mode confusion situations and other potential automation surprises



Autopilot ADEPT model

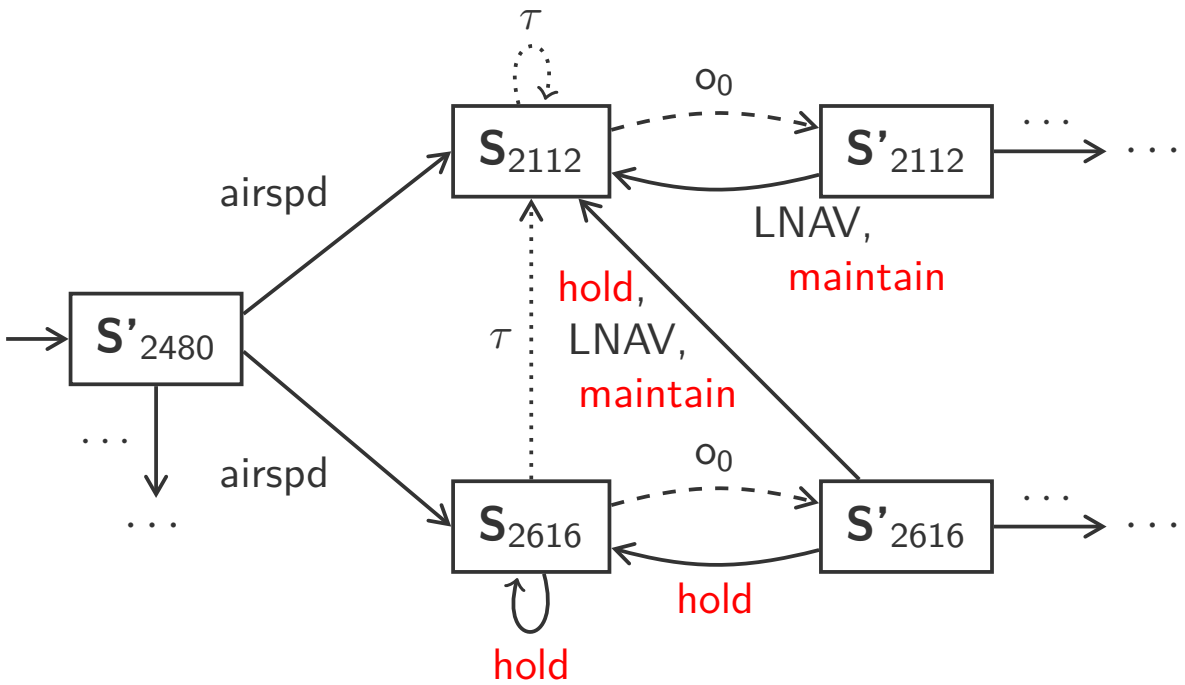


ADEPT models

- Enriched models
  - State-value on the states of the system model
  - Action-guard on the transitions of the mental model
- Formal semantics for ADEPT models
- Translation algorithm towards HVS

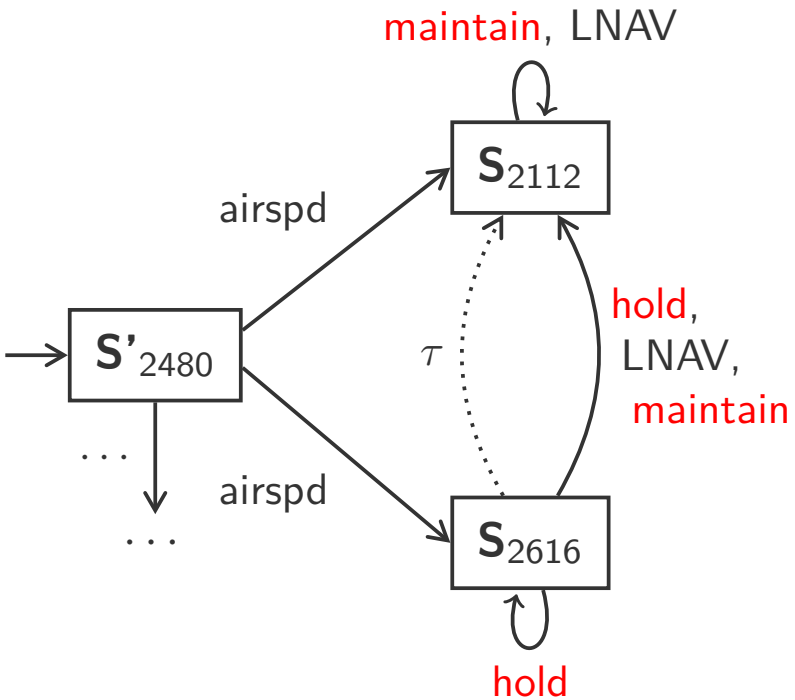
Potential mode confusion

- Potential mode confusion between hold and maintain



Potential mode confusion

- Potential mode confusion between hold and maintain



Contributions

- HMI-LTS, HVS and HVM : translation algorithms
- full-control property, fc-determinism criterion
- Three minimal mental model generation algorithms: 3DFAs, reduction and learning approach
- An analysis framework with a proposed methodology experimentally demonstrated and evaluated
- ADEPT models : semantic, translation algorithm, analysis of an autopilot case study

Future work

- Integrating user tasks models
- Analysing how human error affects full-controllability
- Proposing a precise analysis methodology
- Improving algorithms

Publications

- Sébastien Combéfis, Charles Pecheur. Automatic Generation of Full-Control System Abstraction for Human-Machine Interaction (Formal H 2012)
- Sébastien Combéfis, Dimitra Giannakopoulou, Charles Pecheur, Peter Mehlitz. *A JavaPathfinder Extension to Analyse Human-Machine Interactions* (JPF Workshop 2011)
- Sébastien Combéfis, Dimitra Giannakopoulou, Charles Pecheur, Michael Feary. *Learning System Abstractions for Human Operators* (MALETS 2011)
- Sébastien Combéfis, Dimitra Giannakopoulou, Charles Pecheur, Michael Feary. *A Formal Framework for Design and Analysis of Human-Machine Interaction* (SMC 2011)
- Sébastien Combéfis. *Operational Model: Integrating User Tasks and Environment Information with System Model* (FMIS 2009)
- Sébastien Combéfis, Charles Pecheur. *A Bisimulation-Based Approach to the Analysis of Human-Computer Interaction* (EICS 2009)

Credits

- Naddsy, September 17, 2007, https://commons.wikimedia.org/wiki/File:Airbus_A380_cockpit.jpg.
- Catalina Márquez, January 12, 2007, https://commons.wikimedia.org/wiki/File:Therac-25.jpg.
- Carlos Caballero, May 8, 2019, https://www.carloscaballero.io/software-architecture-therac-25.