LTS-Based Analysis of Interactive Systems

Sébastien Combéfis and Charles Pecheur

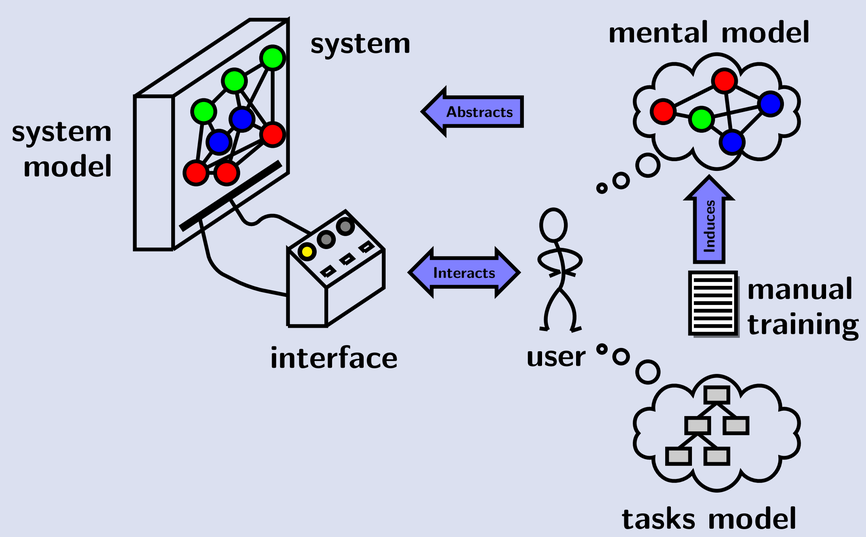
Université catholique de Louvain (UCLouvain, Belgium)

École Polytechnique de Louvain (EPL), Département d'Ingénierie Informatique (INGI)

Place Sainte Barbe 2, 1348 Louvain-la-Neuve

1. Introduction

The aim of the research is to use formal methods to model and reason about interactive systems.



Problems that can be addressed include:

- Check wether a mental model is a good **abstraction** of a given system
- **Generate** a user manual from a given mental model
- **Check** whether user manual allow the user to execute all the user tasks
- Extract part of the system **relevant** to the user regarding a set of user tasks

2. Full-Control Model

Modelling:

- System and mental models as LTSs: $\mathcal{M} = \langle S, \mathcal{L}, s_0, \rightarrow \rangle$
- Action-based user interface: $\mathcal{L}^c$ (commands) $\cup\, \mathcal{L}^o$ (observations) $\cup \{\tau\}$

Problem: Given a system model and an action-based user interface, find the minimal mental model that allows to control the system:

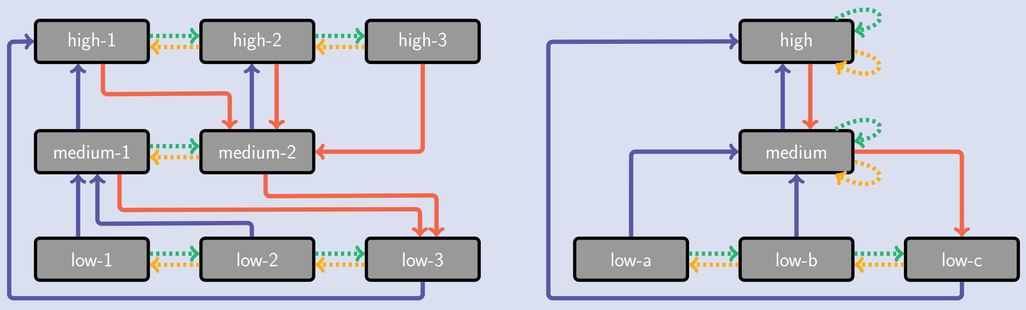
- Commands allowed by mental model are those available on system
- Mental model allows at least observations producible by the system

Approach: Defining an equivalence relation on the system's states, then reducing the system wrt. that equivalence relation. $s \approx_{\text{fc}} t$ iff:

- $\forall s \stackrel{\alpha}{\Longrightarrow} s' : \exists t \stackrel{\alpha}{\Longrightarrow} t' : s' \approx_{\text{fc}} t'$ $\quad (\alpha \in \mathcal{L}^c \text{ a command})$
- $\forall s \stackrel{\beta}{\Longrightarrow} s' : \exists t \stackrel{\beta}{\Longrightarrow} t' \implies s' \approx_{\text{fc}} t'$ $\quad (\beta \in \mathcal{L}^o \text{ an observation})$
- $\forall s \stackrel{\varepsilon}{\Longrightarrow} s' : \exists t \stackrel{\varepsilon}{\Longrightarrow} t' \implies s' \approx_{\text{fc}} t'$ $\quad (\varepsilon \text{ an empty trace})$

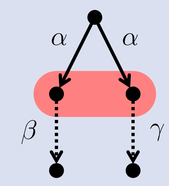
3. Vehicle Transmission System Example

Semi-automatic gearbox (Degani, 2007) **push-up pull-down up down**

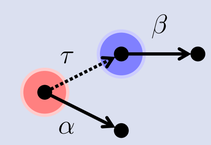


4. Minimization

If the system model is fc-deterministic, i.e. $\forall \sigma : s_0 \stackrel{\sigma}{\Longrightarrow} s \text{ and } s_0 \stackrel{\sigma}{\Longrightarrow} s' \implies s \approx_{\text{fc}} s'$, then $\approx_{\text{fc}}$ gives full-control models.



fc-deterministic



non fc-deterministic

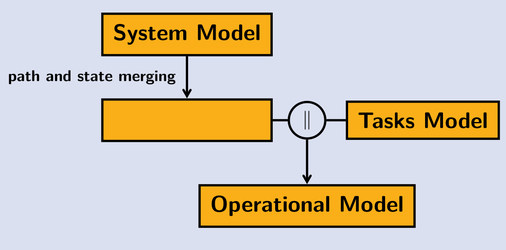
Minimization done with a variant of Paige-Tarjan algorithm.

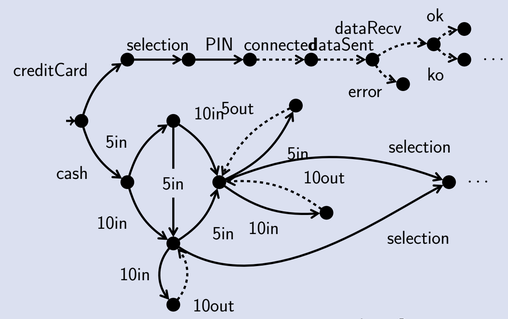
4. Operational Model

Modelling:

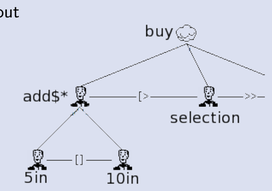
- User's tasks modelleds as a set of LTSs: $\mathcal{T} = \{\mathcal{T}_1, \cdots, \mathcal{T}_k\}$

Problem: Given a system model and tasks model, build an operational model that is a model containing the relevant part of the system regarding the tasks.





- Identifying and merging states of observations path
- Merge states according to tasks hierarchy
- Identifying hub states



5. Conclusion and further work

Contributions:

- Full-control property, equivalence and minimization
- Informal definition of operational model

Further work:

- Mode-preserving model
- Integration information about system's state
- Formal definition of operational model

References

- Sébastien Combéfis, Charles Pecheur. A Bisimulation-Based Approach to the Analysis of Human-Computer Interaction. *EICS2009*, July 2009.
- Sébastien Combéfis. Operational Model: Integrating User Tasks and Environment Information with System Model. *FMIS2009*, November 2009.

MoVES Annual Meeting, December 17, 2009, Université catholique de Louvain, Louvain-la-Neuve, Belgium. Sebastien.Combefis@uclouvain.be