Synthesizing Multi-View Models of Software Systems

Lambeau Bernard

UCL/EPL/INGI

march 2011

Outline

Introduction

A multi-view modeling framework

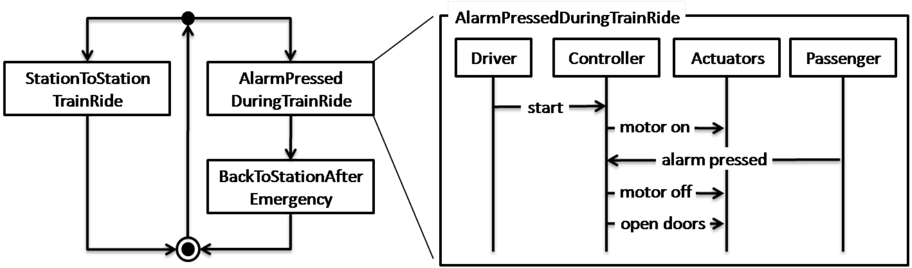
Abstract

Acts as a background chapter about the models we use, their syntax and semantics. Contributions shared with Damas thesis.

Outline

- Event-based Behavior Models
  - Message Sequence Charts (MSC) for instance behaviors
  - Labeled Transition Systems (LTS) for class behaviors
- State-based abstractions
  - Capturing state information with Fluents
  - Guards in behavior models
  - Decorations on behavior models
- Intentional models as goal graphs on fluents
  - Goals and Fluent Linear Temporal Logic (FLTL)
  - Linking FLTL and LTS: property and tester automata

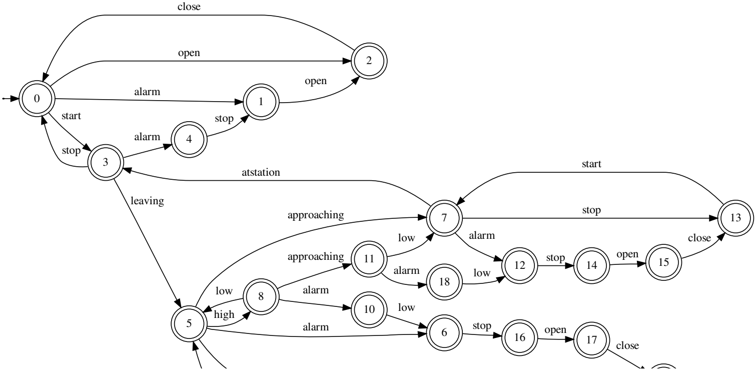
Message Sequence Charts for instance behaviors



MSC (right) and high-level MSC (left)

- Syntax of MSC and hMSC is described in [ITU96]
- Semantics of MSC and hMSC is defined in terms of Labeled Transition Systems, following [UKM03]
- We also allow a hMSC node to be refined as a finer-grained hMSC

Labeled Transition Systems for class behaviors



Labeled Transition Systems

- Syntax and Semantics defined in [MK99]
- Each agent behavior is defined by a LTS. The system behavior is defined by LTS composition
- MSCs are admissible traces in the system LTS [UKM03]

Capturing state information with Fluents

Fluents capture the system state through the occurrence of events [Mil89]

$$fluent\ Fl =< init_{Fl}, term_{Fl} > initially\ Init_{Fl}$$

where $init_{Fl}$ and $term_{Fl}$ are disjoint set of events rendering the fluent *true* and *false*, respectively

Example

$fluent\ moving =< start, \{stop, emergency\ stop\} > initially\ false$
$fluent\ doors\_closed =< close, \{open, emergency\ open\} > initially\ true$

Guards in Behavior Models

Summary

Guards can be formally used in hMSC and LTS, leading to guarded hMSC (g-hMSC) and guarded LTS (g-LTS)

- A guard is a boolean expression on fluents
- Structured forms for hMSC and LTS, avoiding state/trace explosion
- Relax the assumption of fluent initial values being known for all instances

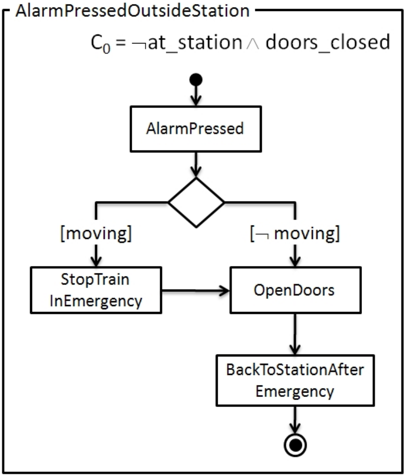
Related publication

Damas C., Lambeau B., Roucoux F. and van Lamsweerde A., *Analyzing Critical Process Models through Behavior Model Synthesis*, in Proc. ICSE'2009: 31th International Conference on Software Engineering, Vancouver, Canada, May 16-24, 2009.

Guards in hMSC, i.e. g-hMSC

Summary

- *Decision nodes*: outgoing transitions are labeled by boolean expressions on fluents
- Initial condition C_0 stating an invariant on the initial state
- Trace semantics through guarded LTS and LTS
- Automated checking of guards: *non overlapping*, *completeness* and *reachability*

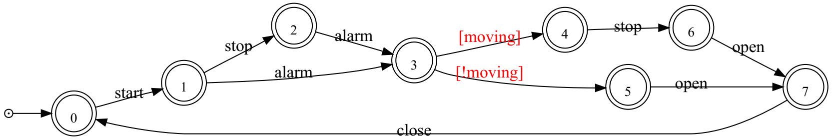


Guards in LTS, i.e. g-LTS

Summary

- A g-LTS transition is labeled by an event or a guard
- Initial condition C_0 stating an invariant on the initial state
- A trace is accepted by a g-LTS if three conditions hold: *trace inclusion*, *admissible start* and *guard satisfaction*

Example



- $C_0 = \neg moving \wedge doors\_closed$
- The event trace (*start alarm open*) is not accepted due to the *guard satisfaction* condition

Deductive synthesis of LTS models from guarded hMSCs

Chapter Outline

- From guarded hMSC to guarded LTS
- From guarded LTS to pure LTS
- Model analysis perspectives of deductive synthesis

References I

[DLvL06] C. Damas, B. Lambeau, and A. van Lamsweerde.
Scenarios, goals, and state machines: a win-win partnership for model synthesis.
In *International ACM Symposium on the Foundations of Software Engineering*, pages 197–207, Portland, Oregon, November 2006.

[ITU96] ITU.
Message sequence charts, recommandation z.120.
International Telecom Union, Telecommunication Standardization Sector, 1996.

[Mil89] R. Milner.
Communication and concurrency.
Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[MK99] J. Magee and J. Kramer.
Concurrency: State Models and Java Programs.
Wiley, 1999.

[UKM03] S. Uchitel, J. Kramer, and J. Magee.
Synthesis of behavorial models from scenarios.
IEEE Transactions on Software Engineering, 29(2):99–115, 2003.