An Optimization Approach for Solving Reachability in Cyber-Physical Systems

Umang Mathur   Atul Sandur

Department of Computer Science

University of Illinois, Urbana Champaign

December 7, 2015

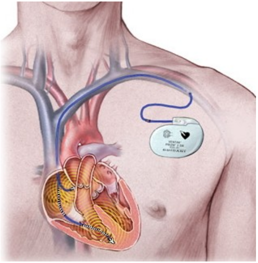
Outline

Cyber-Physical Systems (CPS)

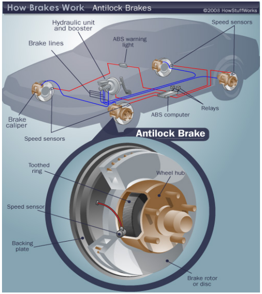
- Cyber-Physical systems are engineered systems that depend upon the integration of
  - computational algorithms, and
  - physical components
- Diverse applications:
  - Healthcare
  - Aerospace, Aeronautics
  - Chemical processes
  - Transportation
  - Energy sector

Cyber-Physical Systems (CPS)

Medical Devices



Avionics



Automobile

Energy

Hybrid Automata: Modelling, Analysis and Synthesis of CPS

- Introduced by Alur et al. to model hybrid systems
- Quite expressive, but undecidable verification (reachability) problems
- Decidable subclasses exists, e.g.
  - Timed Automata (Alur, and Dill),
  - Initialized Rectangular Hybrid automata (Henzinger et al.),
- Most verification techniques rely on exhaustive exploration of state space using finite bisimulations

- Introduced by Alur et al. to model hybrid systems
- Quite expressive, but undecidable verification (reachability) problems
- Decidable subclasses exists, e.g.
  - Timed Automata (Alur, and Dill),
  - Initialized Rectangular Hybrid automata (Henzinger et al.),
- Most verification techniques rely on exhaustive exploration of state space using finite bisimulations

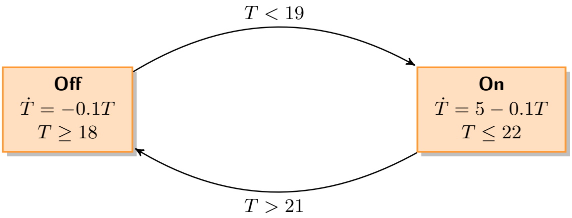


Figure : Modelling a smart heater as a Hybrid Automata

Introduction

Verification and Testing of Hybrid Automata

Example

Singular Hybrid Automata: Syntax and Semantics

Reachability

Concolic walk

Our approach

Future Work

Reachability in Hybrid Systems

Safety Critical Systems :

- Nuclear reactors
- Chemical plants
- Aeronautics/Automobiles

It is therefore important to have certain safety guarantees for such systems

Checking reachability of certain states, thus, is a natural question to ask

- Can reach some error state ?
- How to reach ?
  - input ?
  - path ? (non-determinism)

Other interesting applications:

- Motion planning

Robotic Motion Planning

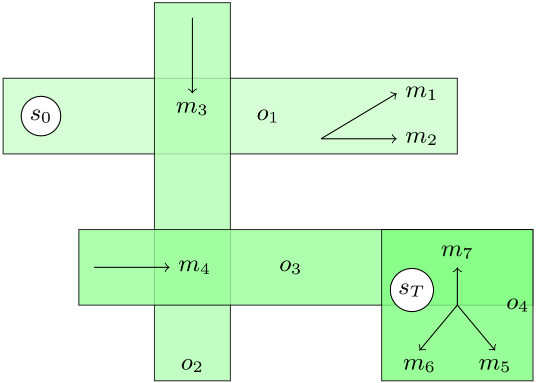


Figure : Robotic motion planning problem modelled as a reachability question

– Can a bot enter $o4$ starting from some point in region $o1$

Syntax of SHA

Syntax : Singular Hybrid Automata (SHA)

A singular hybrid automaton is a tuple $\mathcal{H} = (M, M_0, \Sigma, X, \Delta, I, F)$ where

- M is a finite set of control modes and $M_0 \subseteq M$,
- Σ is a finite set of actions,
- X is an (ordered) set of variables,
- $\Delta \subseteq M \times \text{poly}(X) \times \Sigma \times 2^X \times M$ is the transition relation,
- $I : M \to \text{poly}(X)$ is the mode-invariant function, and
- $F : M \to \mathbb{Q}^{|X|}$ is the mode-dependent flow function characterizing the rate of each variable in each mode.

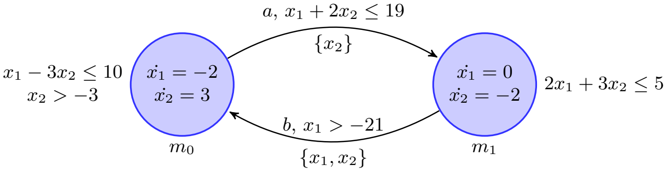


Figure : Example SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

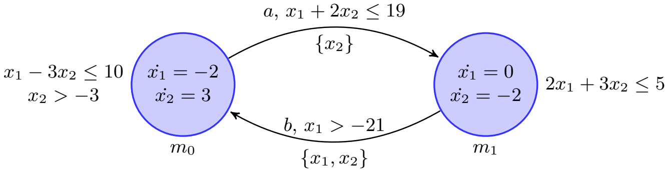


Figure : Example run in a SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

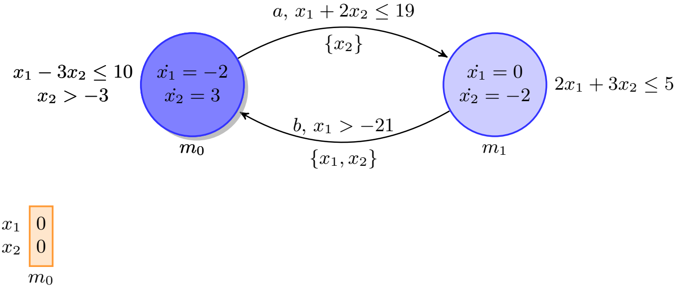


Figure : Example run in a SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

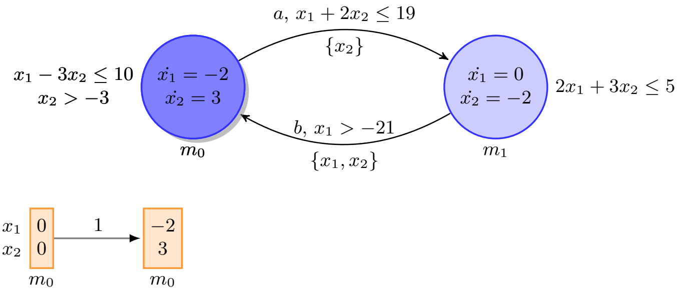


Figure : Example run in a SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

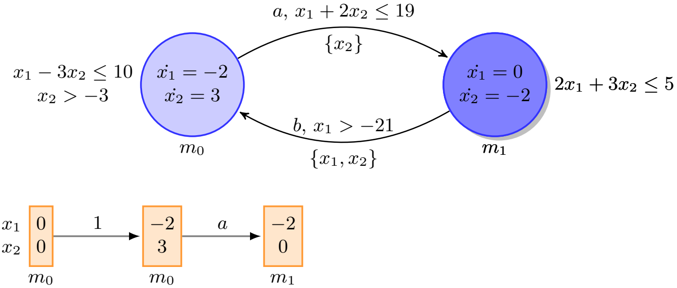


Figure : Example run in a SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

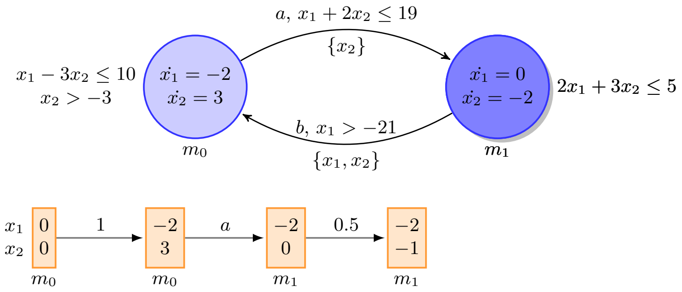


Figure : Example run in a SHA

Semantics of SHA

- Configuration (m, ν), $m \in M$, $\nu \in \mathbb{R}^{|X|}$
- Timed action (t, a), $t \in \mathbb{R}^{\geq 0}$ and $a \in \Sigma$
- Transition $((m, \nu)(t, a)(m', \nu'))$
- A run is a sequence of transitions $(m_0, \nu_0)(t_1, a_1)(m_1, \nu_1)(t_2, a_2) \cdots$

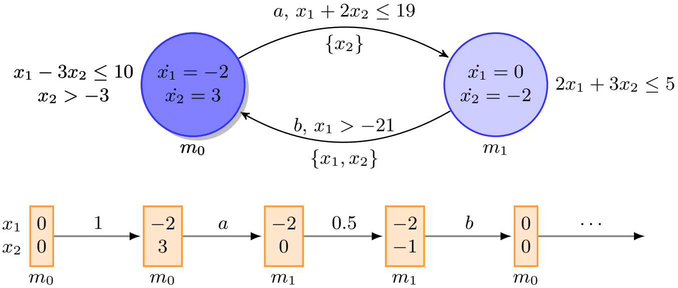


Figure : Example run in a SHA

Modelling Robot Motion Planning Using SHA

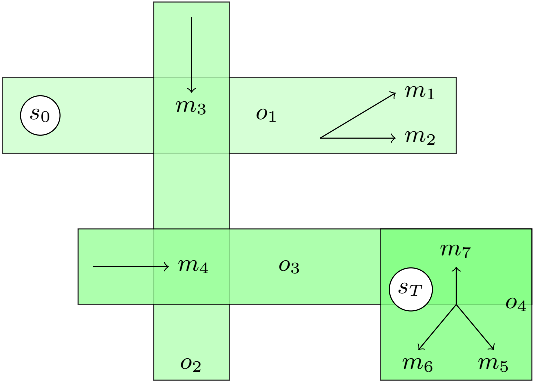


Figure : Robotic motion planning problem: Modelling as a SHA

Modelling Robot Motion Planning Using SHA

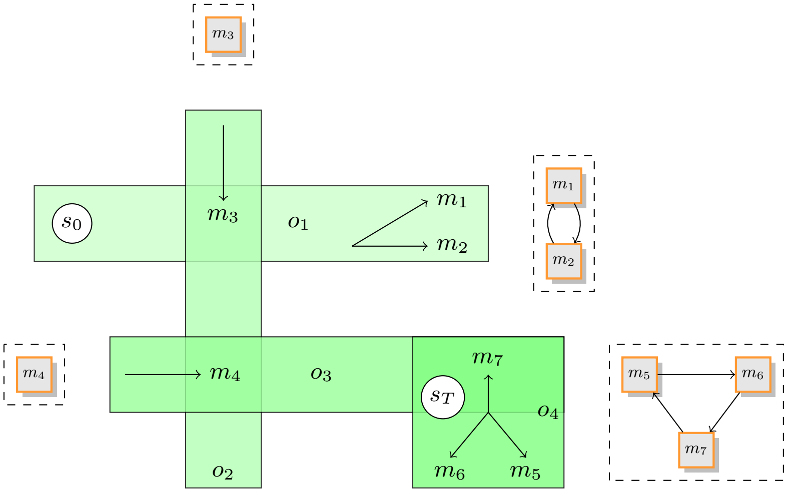


Figure : Robotic motion planning problem: Modelling as a SHA

Modelling Robot Motion Planning Using SHA

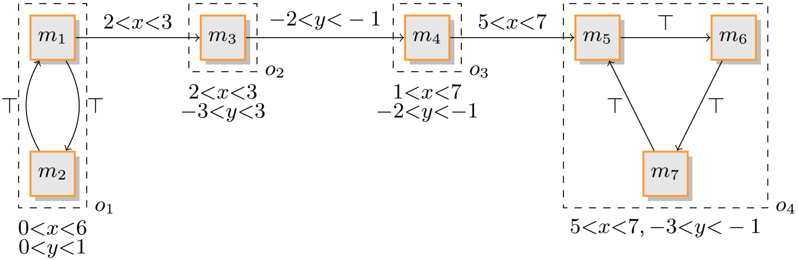


Figure : Singular Hybrid Automaton for robotic motion planning example

Reachability in SHA

Configuration Reachability Problem

Given a singular hybrid automaton $\mathcal{A}$, a set of starting configurations $\mathcal{S}$, and a set of target configurations $\mathcal{T}$, decide whether there exists a

- finite run
- starting from some starting from some $(m, \nu) \in \mathcal{S}$, and
- ending in some $(m', \nu') \in \mathcal{T}$

Reachability in SHA

Configuration Reachability Problem

Given a singular hybrid automaton $\mathcal{A}$, a set of starting configurations $\mathcal{S}$, and a set of target configurations $\mathcal{T}$, decide whether there exists a

- finite run
- starting from some starting from some $(m, \nu) \in \mathcal{S}$, and
- ending in some $(m', \nu') \in \mathcal{T}$

Theorem (Henzinger et. al., '98)

Configuration reachability problem is undecidable for 3 or more continuous variables.

Concolic walk

Concepts

- Combination of symbolic reasoning, concrete evaluation and heuristic search
- Use fitness function to measure how close a point in half-space (obtained from linear constraints) is to global solutions for whole path condition
- Search heuristic uses fitness function to guide random walk towards promising regions in valuation space
- Not complete, but sound approach to handle non-linear constraints in path condition

Our approach

Concepts

- Combination of symbolic reasoning, concrete evaluation and heuristic search
- Use fitness function to measure how close a point in half-space (obtained from linear constraints) is to global solutions for whole path condition
- Search heuristic uses fitness function to guide random walk towards promising regions in valuation space
- Not complete, but sound approach to handle non-linear constraints in path condition

Summary and Future Work

– Future work

Thank You !