Compositional and Distributed Verification of Distributed Hybrid Systems

David Renshaw, Ping Hou, André Platzer, Sarah M. Loos
Computer Science Department, Carnegie Mellon University, Pittsburgh, PA

Overview

Issue: Distributed hybrid systems present extraordinarily challenging problems for verification. On top of the difficulties associated with distributed systems, they also exhibit continuous dynamics. Handling the arithmetic challenges can be extremely expensive. How do we succinctly specify a compositional and distributed verification strategy, and how do we control the computational cost?

Objective: Develop a compositional and distributed verification tool for distributed hybrid systems, which has a distributed verification engine. Using a distributed verifying backend enables us to overcome the high computational complexity of distributed hybrid systems verification.

Distributed Hybrid Systems Verification

Discrete, Continuous, Distributed:

Initially safe → \([\text{step}]\) No collision
step: free dynamics; discrete coordination; coordinated dynamics
continuous dynamics (quantified differential equation): \(\forall x(i)' = a(i)\)
discrete dynamics (quantified control decisions): \(\forall d(i) = \text{if.. then } a \text{ else } b \text{ fi}\)
dimensional dynamics (appearance): \(\forall i \Rightarrow \text{new Aircraft}\)

Continuous dynamics of an aircraft \(i\) with an angular velocity \(\omega(i)\):
\[F_{\omega(i)}(i): x_1(i)' = d_1(i), x_2(i)' = d_2(i), d_1(i)' = -\omega(i)d_2(i), d_2(i)' = \omega(i)d_1(i)\]
 Continuous dynamics of all aircraft \(i\):
\(\forall i(F_{\omega(i)}(i))\)

Differential Invariants:

\(\neg F\quad F\quad\neg F\quad F\)
Quantified differential invariant

Verification in KeYmaeraD:

Verifying state
Decompose subproblems
Or-branching for choices and decomposition
Closing an or-branching

KeYmaera and KeYmaeraD:

KeYmaera
KeYmaeraD

The cost of verification is concentrated in the tree's leaves

Case Study:

Distributed Highway Control

Verification Statistics:

synchronous control:
verification state: 1134 nodes
one worker time: 40 seconds
two workers time: 33 seconds
2.86GHz Core 2 Duo

asynchronous control:
verification state: 7154 nodes
one worker time: 640 seconds
four workers time: 195 seconds
2.83GHz intel Core2 quad core

Initial safety → \([\text{step}]\) No collision
step: exit \(\cup\) enter \(\cup\) (control; dynamics)

KeYmaeraD tool architecture

Initially safe → \([\text{step}]\) No collision
invariant → \([\text{step}]\) invariant
step: exit \(\cup\) enter \(\cup\) (control; dynamics)

Verification Statistics:

synchronous control:
verification state: 1134 nodes
one worker time: 40 seconds
two workers time: 33 seconds
2.86GHz Core 2 Duo

asynchronous control:
verification state: 7154 nodes
one worker time: 640 seconds
four workers time: 195 seconds
2.83GHz intel Core2 quad core