Logic and Hybrid Systems

Manasvi Saxena

Formal Systems Lab, UIUC

Hybrid Systems

- Dynamical Systems exhibiting both discrete (jump) and continuous (flow) behaviors.
- Serve as models of physical systems, from thermostats to trains.
- Continuous dynamics specified using Differential Equations.

Differential Dynamic Logic (dL)

- Main focus - Differential Dynamic Logic for Hybrid Systems (Andre Platzer).

Differential Dynamic Logic (dL)

- Main focus - Differential Dynamic Logic for Hybrid Systems (Andre Platzer).
- Practical deductive verification of hybrid systems.

Differential Dynamic Logic (dL)

- Main focus - Differential Dynamic Logic for Hybrid Systems (Andre Platzer).
- Practical deductive verification of hybrid systems.
- Introduces Hybrid Program - program notation for hybrid systems.

Differential Dynamic Logic (dL)

- Main focus - Differential Dynamic Logic for Hybrid Systems (Andre Platzer).
- Practical deductive verification of hybrid systems.
- Introduces Hybrid Program - program notation for hybrid systems.
- Dynamic Logic for Hybrid Programs, a generalization of Dynamic Logic.

Differential Dynamic Logic (dL)

- Main focus - Differential Dynamic Logic for Hybrid Systems (Andre Platzer).
- Practical deductive verification of hybrid systems.
- Introduces Hybrid Program - program notation for hybrid systems.
- Dynamic Logic for Hybrid Programs, a generalization of Dynamic Logic.
- Suited for automation.

Hybrid Automata

- Commonly used to model Hybrid Systems, via Graphs.
- Nodes specify continuous dynamics. Edges describe discrete transitions.
- Intuitive, but not suitable for deductive verification.

Hybrid Automata

- Commonly used to model Hybrid Systems, via Graphs.
- Nodes specify continuous dynamics. Edges describe discrete transitions.
- Intuitive, but not suitable for deductive verification.

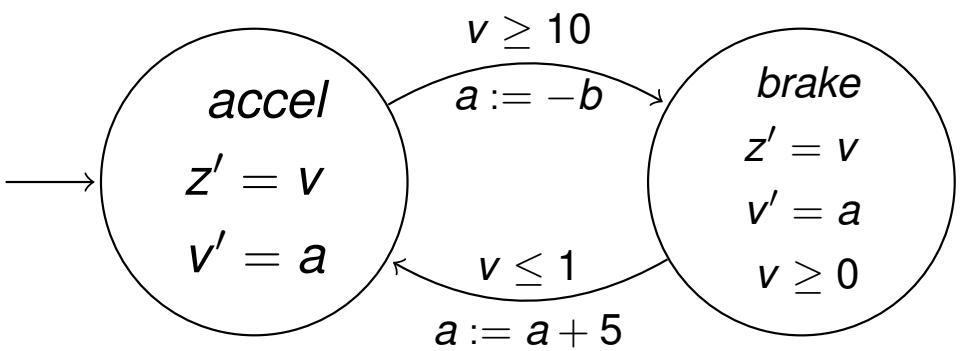

Figure: Hybrid Automata (simplified) of a Train Control System

Differential Dynamic Logic

Motivations

- **First Order Logic** - No builtin means for referring to state transitions.
- **Temporal Logics** - Modal operators allow referring to state transitions. But valid formulas only express generic facts.

Differential Dynamic Logic

Motivations

- **First Order Logic** - No builtin means for referring to state transitions.
- **Temporal Logics** - Modal operators allow referring to state transitions. But valid formulas only express generic facts.
- **Dynamic Logic (DL)** - Combines operational system models with operators for reasoning.
  - Provides parameterized modal operators, $[\alpha]$, $\langle\alpha\rangle$ that refer to states reachable by system α.
  - $[\alpha]\phi$ expresses all states reachable by α satisfy ϕ, allowing reasoning about discrete systems.
  - Say $(b > 0) \rightarrow [a := -b](a < 0)$ expresses a discrete transition. Using DL's calculus, we get $(b > 0) \vdash (a < 0)[b/a]$. Convenient for reasoning about discrete behavior.

Differential Dynamic Logic

Motivations

- **First Order Logic** - No builtin means for referring to state transitions.
- **Temporal Logics** - Modal operators allow referring to state transitions. But valid formulas only express generic facts.
- **Dynamic Logic (DL)** - Combines operational system models with operators for reasoning.
  - Provides parameterized modal operators, $[\alpha]$, $\langle\alpha\rangle$ that refer to states reachable by system α.
  - $[\alpha]\phi$ expresses all states reachable by α satisfy ϕ, allowing reasoning about discrete systems.
  - Say $(b > 0) \rightarrow [a := -b](a < 0)$ expresses a discrete transition. Using DL's calculus, we get $(b > 0) \vdash (a < 0)[b/a]$. Convenient for reasoning about discrete behavior.
  - No built in notion for describing or reasoning about continuous dynamics.

Differential Dynamic Logic

Motivations

- Generalize DL so operational models α can be used in modal formulas like $[\alpha]\phi$. dL refers to generalized models as “Hybrid Programs”.
- A compositional calculus for verification. Decompose $[\alpha]\phi$ into an equivalent formula $[\alpha_1]\phi_1 \wedge [\alpha_2]\phi_2$.
- Prove subsystems and subproperties $[\alpha_i]\phi_i$ independently and combine results conjuntively.
- Complete relative to handling of differential equations.

Differential Dynamic Logic

Syntax and Semantics

dL formulas built over V, set of real-valued logical variables and signature Σ containing functions, predicate symbols over reals, like $0, 1, +, \geq$.

Σ also contains *System State Variables*. Unlike rigid symbols, like $1, 2$, their interpretation can change from state to state.