The Automatic Analysis of Hybrid Systems

R. Alur et al

Outline

- Formal specification of hybrid systems
- Classifications of hybrid systems
- Computability results
- Verification in terms of reachability

Formal specification of hybrid systems

A hybrid system is modeled as a finite automata, equipped with a set of real variables and each location has evolution laws constraining the values of the variables over time. The state of the automaton is a $(location, valuation)$ pair.

Formal specification cont'd

- $H = (Loc, Var, Lab, Edg, Act, Inv)$
  - Loc - locations, i.e. modes of the system, states in the discrete automata
  - Var - variables
  - Lab - synchronization labels used in parallel composition
  - Edg - edges in the graph, transitions (l, a, μ, l') with label a and a rule for variable re-assignments, the transition relation μ
  - Act - activities associated with locations, the (time-invariant) laws of variable evolution
  - Inv - invariants for each location, describing the possible valuations

Formal specification further cont'd

A run of a hybrid system is a sequence of states interleaved with timed activities.

1. The starting point of the activity is the valuation of the state and
2. the activity evolves according the invariant of the location.
3. A successor state in the sequence corresponds to a target location of an edge with the current state as source state.

Getting tired of Formal specifications?

Labeled transition systems (LTS) is a well studied concept when discussing formal semantics and behavior of (discrete) systems. An LTS is a triple $(states, lables, transitions)$ where the labels define action names and transitions are $(state, label, state)$ triples. The LTS for a hybrid system is defined by two rules.

1. Transitions steps - if there is an edge with appropriate transition relation, constrained by the invariant, we allow a transition between locations labeled by the synchronization label of the edge
2. Time steps - Staying in the same location, we allow the variables to evolve according to the corresponding actions and invariants.

Last formal spec slide, I promise!

Parallel composition $H = H_1 \times H_2$ of two hybrid systems allow the systems to interact by synchronization on common synchronization labels. The product hybrid system H is forbidden to take transitions on labels that occur in both H_1 and H_2, if not both H_1 and H_2 does so simultaneously.

Linear Hybrid Systems

$$\frac{dx}{dt} = k$$

Transition relations are guarded assignments, e.g. $v(guard) \wedge \forall x \in Var.v(\alpha_x) \leq v'(x) \leq v(\beta_x)$

Classifications of linear hybrid systems

- Discrete systems - $Act(l, x) = 0$, variables change only on transitions
- Finite-state systems - all variables are propositions, $\mu(e, x) \in \{0, 1\}$ for all $e \in Edg$
- Timed automatons - all variables are propositions or clocks ($Act(l, x) = 1$,$\mu(e, x) \in \{0, x\}$), and linear expressions are boolean combinations of inequalities
- Multi-rate timed systems - all variables are propositions or skewed clocks ($Act(l, x) = k_x$,$\mu(e, x) \in \{0, x\}$)
- Integrator systems - all variables are propositions or integrators ($Act(l, x) \in \{0, 1\}$,$\mu(e, x) \in \{0, x\}$).

Computability results

- Reachability is decidable for simple multi-rate timed systems - proven by skewing back the clocks and obtaining a timed automata.
- Reachability is undecidable
  - for 2-rate timed systems - use a 2-rate timed system to encode a 2-counter machine
  - for simple integrator systems

Verification in terms of reachability

- Forward analysis $X = \langle I \cup post[X] \rangle \nearrow$
- Backward analysis $X = \langle R \cup pre[X] \rangle \swarrow$
- Approximate analysis
  - Abstract interpretation, convex hull, widening
- Minimization
  - compute minimal bisimulation

Questions

- Do you think there is any relation between the concepts of reachable region and controllability?
- I would be extremely interested to hear a control engineer opinion if the decidable classes of hybrid systems, presented in the paper, have the necessary modeling power for complicated control applications?
- Why is the members of the set Lab called synchronization labels? Why are they needed?
- What do components 5 and 6 really mean in the definition of a hybrid system (page 5)? (Especially the concepts time-invariant, and time-deterministic)
- In the linear hybrid systems, why cannot k_x be a real number?