Formal Techniques for Analyzing Hybrid Systems

Ashish Tiwari

Motivation

We want to mathematically model a system and analyze its behavior using that model

- Systems such as cars, drones, airplanes that have both a cyber component (programs, protocols, controllers, planners, etc.) and a physical component

Most of modern science and engineering uses modeling and analysis

We, however, want to analyze things a bit more formally

There is plenty of motivation for why we want to model and analyze dynamical systems

- predict what will happen, increase confidence about behavior, establish correctness, reduce costs, etc.

Dynamical Systems

A dynamical system consists of a notion of

- state space X – defines the configurations of the system
- collection of traces $F : \mathsf{T} \mapsto \mathsf{X}$ – defines all possible evolutions of the system, for some notion T of time

The state space X is usually described as the set of valuations of a given set of variables.

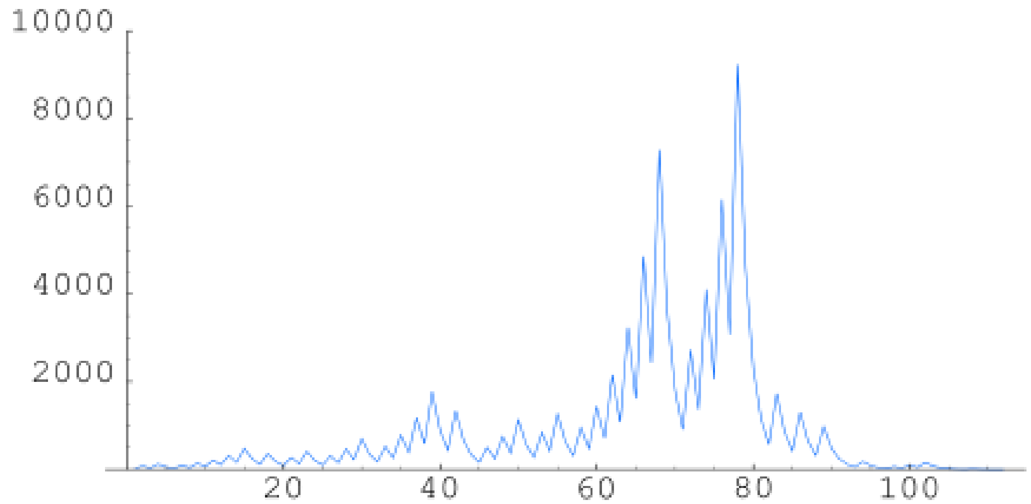
Dynamical system: Example

Example: Consider one variable x taking values in the set of positive integers.

How to define the dynamics?

Rule: $x(t+1) = x(t)/2$ if $x(t)$ is even, else $3 * x(t) + 1$

A trace $F(t)$ of the system when initially $F(0) = 27$ is shown below.



Dynamical Systems: Classifying based on X and T

We can have different kinds of state space:

- discrete-space: $\mathsf{X} \subset \mathbb{N}^n$
- continuous-space: $\mathsf{X} \subset \mathbb{R}^n$
- hybrid-space: $\mathsf{X} \subset \mathbb{N}^n \times \mathbb{R}^m$

We can have different kinds of "time":

- discrete-time: $\mathsf{T} = \mathbb{N}$
- continuous-time: $\mathsf{T} = \mathbb{R}^{\geq 0}$
- hybrid-time: $\mathsf{T} = \mathbb{R}^{\geq 0} \times \mathbb{N}$

The example above was a discrete-time discrete-space dynamical system

Dynamical System: Exercise

1. Construct an example of a discrete-time continuous-space dynamical system
   - Newton's method for finding a root of f:
   $$x(t+1) = x(t) - f(x(t))/f'(x(t))$$
2. Construct an example of a continuous-time continuous-space dynamical system
   - Freely falling ball:
   $$dy(t)/dt = v(t), \qquad dv(t)/dt = -9.81$$

Question: What is the state space of these two systems?

Dynamical System: Finite- and Infinite-State

A system is finite-state if there are finitely many elements in its state space $\mathbf{X}$.

It is infinite-state otherwise.

- All continuous-space systems are infinite-state.
- A dynamical system defined by a finite-state automaton is finite-state.

Dynamical Systems: Classifying based on Dynamics

A system is deterministic if fixing $F(0)$ fixes F

Example: The $3n+1$ system was deterministic

In a nondeterministic system, the same state can have multiple "successors".

Sometimes we assign a probability to each of the different options available at a state
This gives us a stochastic dynamical system.
Eg. Markov chain

Dynamical Systems: Open and Closed

A system is open if its dynamics F is influenced by some input variables.

A system is closed if it has no input variables

The input variables can range over some predefined set of values

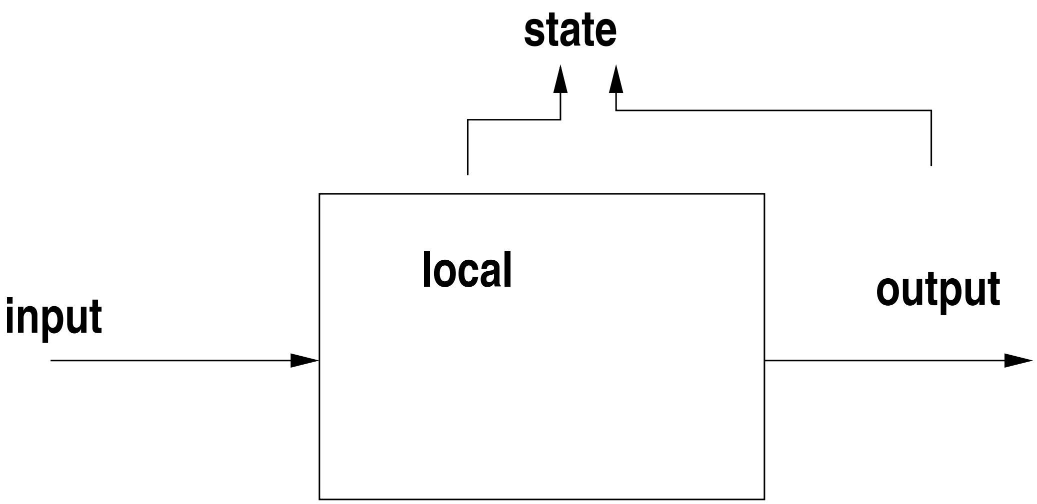
The input variables are disjoint from the state variables

Example: A human, or a controller, turns a heater on or off. The dynamics of the room temperature changes based on this Boolean input.

Dynamical Systems: Observables

Some part of the state space of a system might be externally observable

This is modeled by defining a set of output variables, which is a subset of the state variables



How to describe a dynamical system?

Recall the two parts of a dynamical system:

- State space X: we can just list the variables and their types
  - we can identify local and output variables, and additional input variables if any
- Collection of temporal traces: We write rules that tell us how the system evolves locally
  - a program
  - a differential equation
  - a collection of rules
  - compose smaller dynamical systems

Examples

Spring-mass system:

- State space: $\mathbb{R}^2$, valuations of x, v
- Dynamics: $\frac{dx}{dt} = v, \quad \frac{dv}{dt} = -x$

A sorting program:

- State space: $\mathbb{Z}^n$, valuations of $x_1, \ldots, x_n$
- Dynamics: for any i, j: swap x_i, x_j if $x_i > x_j$

Exercise: Classify these two dynamical systems. open/closed? det/non-det/stochastic? finite-state? discrete-time? discrete-space?

Summary So Far

We defined dynamical systems, and the various classes of such systems

We saw a few examples

We discussed how to write a dynamical system

Next, we want to analyze dynamical systems to see if they have some desired property.

How do we specify the desired properties?

Property Specification Language

Logic! To say things about temporal behavior, we have temporal logics.

$$\phi := \top \mid at \mid \neg\phi \mid \phi \vee \phi \mid \mathbb{F}\phi \mid \mathbb{G}\phi \mid \phi \mathbb{U} \phi$$

Here at denotes an atomic formula that can be evaluated on a state; e.g. $x < 20$

What does a formula mean? It is a property of traces, and hence we should know if a given trace satisfies a given formula.

Semantics for our Temporal Logic

$$
\begin{aligned}
F &\models \top \\
F &\models at \quad \text{if } at \text{ is true at state } F(0) \\
F &\models \neg\phi \quad \text{if } F \not\models \phi \\
F &\models \phi_1 \vee \phi_2 \quad \text{if } F \models \phi_1 \text{ or } F \models \phi_2 \\
F &\models \mathbb{F}\phi \quad \text{if } \exists t \geq 0 : F(t :) \models \phi \\
F &\models \mathbb{G}\phi \quad \text{if } \forall t \geq 0 : F(t :) \models \phi \\
F &\models \phi_i \mathbb{U} \phi_2 \quad \text{if } \exists t \geq 0 : F(t :) \models \phi_2 \wedge \forall (t_1 < t) : F(t_1 :) \models \phi_1
\end{aligned}
$$

The notation $F(t :)$ denotes a suffix-trajectory of F that starts at state $F(t)$
The time domain $\mathbf{T}$ is assumed unbounded

Temporal Properties

$\mathbb{F}$ operator says that eventually something happens

- $\mathbb{F}(good)$: eventually something good happens

$\mathbb{G}$ operator says that always something holds

- $\mathbb{G}(notbad)$: nothing bad ever happens

Relationship between temporal operators:

$$\begin{aligned} \mathbb{G}(\phi) &= \neg\mathbb{F}(\neg\phi) \\ \mathbb{F}(\phi) &= \top\mathbb{U}\phi \end{aligned}$$

Deadlock and Zeno

Recall: The time domain $\mathbf{T}$ is assumed unbounded

If a model deadlocks, then time ceases to progress

A hybrid system could have zeno behaviors.

Example: Bouncing ball

Reachable deadlock states, or zeno behaviors, are undesirable

Detecting if a system has such behavior is difficult

Initial States

Before we can ask if a dynamical system satisfies a property, we need one last thing

We need to specify a set of initial states $I \subseteq \mathsf{X}$

We are only interested in trajectories F s.t. $F(0) \in I$

The Model Checking Problem

Given a dynamical system $S = (\mathsf{X}, \mathsf{F}, I)$

- X - set of states
- F - collection of trajectories
- I - set of initial states

And given a temporal property ϕ (whose atomic formulas get evaluated on X determine if

$$F \models \phi$$

for every $F \in \mathsf{F}$ s.t. $F(0) \in I$

Notation: $S \models \phi$

Model Checking Problem: Example

Let $S = (\mathbb{N}^{\{x\}}, \mathsf{F}, \mathbb{N}^{\{x\}})$ be the $3n + 1$ system

An example of the model checking problem:

Is it the case that

$$S \models \mathbb{F}(x == 1)$$

This is an open problem

Model Checking Problem: Example

For the same system S, consider:

- $S \models (x < 5) \Rightarrow \mathbb{G}(x < 15)$?
  - No
  - Counter example: The trace in F where formula evaluates to false
  - $3, 10, 5, 16, 8, (4, 2, 1)^*$
- $S \models (x < 5) \Rightarrow \mathbb{G}(x \leq 16)$?
  - Yes

Summary

- Dynamical system: $(\mathsf{X}, \mathsf{F}, I)$
  - each trajectory in $F \in \mathsf{F}$ is a mapping from T to X
- Property language - a temporal logic
  - a ϕ is either true or false in any given trajectory
- Model checking problem:
  - Determine if $S \models \phi$

Tool Install

- Download and install SAL (executable)
  `http://sal.csl.sri.com`
- Install HybridSal relational abstractor:
  `http://www.csl.sri.com/users/tiwari/relational-abstraction/`