Under-Determined Dynamical Systems

Oded Maler

CNRS - VERIMAG
Grenoble, France

FAC Workshop 2011

Disclaimer

- I am a theoretician
- I do not have to ship a chip or deliver software in time for market ...
- ... or pretend that I have to
- So I have time to sit and contemplate on **trivial** but **general** abstract models..
- ... free from the messy details of the concrete instances
- Sometimes it is useful – sometimes not

Summary

- By **under-determined** dynamical systems: dynamical systems where not all the details have been filled out
- Systems for which you have to provide additional information in order to run a **simulation**
- This information is taken from some **uncertainty space** (or **ignorance space**)
- We make distinction between **static** (punctual) and **dynamic** under-determination
- Simulation, testing, verification, monte-carlo, parameter-space exploration are all different ways to take this uncertainty space into account

Dynamical Models

- State space X, say a bounded subset of $\mathbb{R}^n$ (or $\mathbb{B}^n$ for discrete systems)
- Behavior, run, trajectory, **trace**:

$$x[t_0], x[t_1], x[t_2], \ldots$$

- What does a simulator need to produce such a trace?
- For **deterministic** systems the dynamic rule is a function $f : X \to X$
- (Hopefully f represents faithfully the phenomenon/system we are interested in)
- The rule allows the simulator to proceed from one state to another

$$x[t_{i+1}] = f(x[t_i])$$

- You just have to **fix** the initial state $x[0]$

Static/Punctual Under-Determination

- Some systems may have a unique initial state (computer people like to reboot..)
- Hence in the deterministic case they can immediately produce a (unique) trace
- For other systems, you need to pick x_0 from some subset X_0 that contains all conceivable initial states
- Without this information, the system has an empty slot that needs to be filled by some x
- In this sense the system without this information is under-determined and cannot generate a trace
- The missing item is a point in $X = \mathbb{R}^n$, that should be determined before we produce the trace

Models and Reality

- Whenever our models are supposed to represent something non-trivial they are just **approximations**
- This is evident for anybody working in concrete **physical** systems
- Especially systems where the material realization technology has not been fixed
- Somewhat less so for those coming from functional verification of digital hardware or software
- One common way to pack our ignorance in a compact way is to introduce **parameters** ranging in some **parameter space**

Examples:

- **Biochemical reactions** in the cells following the **mass action** law
- Many parameters related to the affinity between molecules
- Cannot be deduced from first principles, only measured by isolated experiments under different conditions
- **Timing performance analysis** of a new application (task graph) on a new multi-core architecture
- Execution times of tasks not known before the application is written and the architecture is developed
- **Voltage level** modeling and simulation of circuits
- A lot of variability in transistor characteristics depending on production batch, place in the chip, temperature, etc.

Parameterize Dynamical System

- The dynamics f becomes a **template** with some empty slots to be filled by parameters
- Taken from some parameter space $P \subseteq \mathbb{R}^m$
- Each p instantiates f into a concrete function f_p that can be used to produce traces
- Parameters like initial states are instance of punctual under-determination: you choose them only once when starting the simulation
- In fact, you can add the parameters as state variables that do not change
- Let $X' = X \times P$ and define on it dynamics f' as

$$f'(x, p) = (f_p(x), p)$$

- As if at the beginning the transistor sees where it is and what dynamics it must follow

So?

- So you have a model which is under-determined, or equivalently an infinite number of models
- For simulation you need to determine, to make a choice to pick a point p in the parameter space
- The simulation shows you something about **a** possible behavior of the system
- But it could be otherwise with another choice
- Ho do you live with that?

Possible Attitudes

- The answer depends on many factors
- One is the responsibility of the modeler/simulator- what are the consequences of not taking under-determination seriously
- Another factor is the mathematical and real natures of the system you are dealing with
- And as usual, it may depend on culture, background and tradition

Non Responsibility: a Cartoon

- Suppose you are a scientist not engineer, say biologists
- You conduct experiments and observe traces
- You propose a model and start tuning the parameters until you obtain a trace similar to the one observed experimentally
- This are **nominal** values of the parameters
- Then you can publish a paper about it
- Unless you have picky reviewers that check robustness, there are not much consequences if you neglect under-determination
- The situation is different if some engineering is involved (pharmacokinetics, synthetic biology) or if you want to compose models

Justified Nominal Value

- You can get away with using a nominal value if your system is very continuous and and well-behaving
- Then points in the neighborhood of p generate similar traces
- There are also mathematical techniques (bifurcation diagrams, etc.) that can tell you sometimes what happens when you change parameters
- This continuity can be easily broken by mode switching
- Another justification for ignoring parameter variability is if the system is adaptive anyway to deviations from nominal behavior (control, feedback)

Taking Under-Determination More Seriously: I

- Paranoid worst-case formal verification attitude:
- If we say something about of the it should be provably true for all choices of p
- Instead of doing a simple simulation you do **set-based** simulation computing **tubes of trajectories** covering everything
- Advantages: works also for hybrid (switched) systems, can handle dynamic uncertainty
- Limitations: have to manipulate geometric objects in high dimension
- State-of-the-art: linear and piecewise-linear dynamics > 200 state variables; Nonlinear: 10-20 variables

Taking Under-Determination More Seriously: II

- One can **sample** the parameter space with or without probabilistic assumptions
- Make a grid (exponential in the number of parameters) or throw a coin
- We developed a method for intelligent search in the parameter space
- Sensitivity information from the numerical simulator can tell you at the end of the simulation whether you need to refine the coverage of the parameter space
- Arbitrary dimensionality of the state space, but no miracles against the dimensionality parameter space

Dynamic Under-Determination

- The system is **open**, exposed to external disturbances
- Dynamics of the form

$$x[t_{i+1}] = f(x[t_i], v[t_i])$$

- Now the under-determination is dynamic
- To produce a trace you need to give the value of v at every step in time
- The most appropriate way to represent the influence of other unmodeled subsystems and the external environment
- Again, it can be some nominal value: step response, periodic signal, random noise, etc.