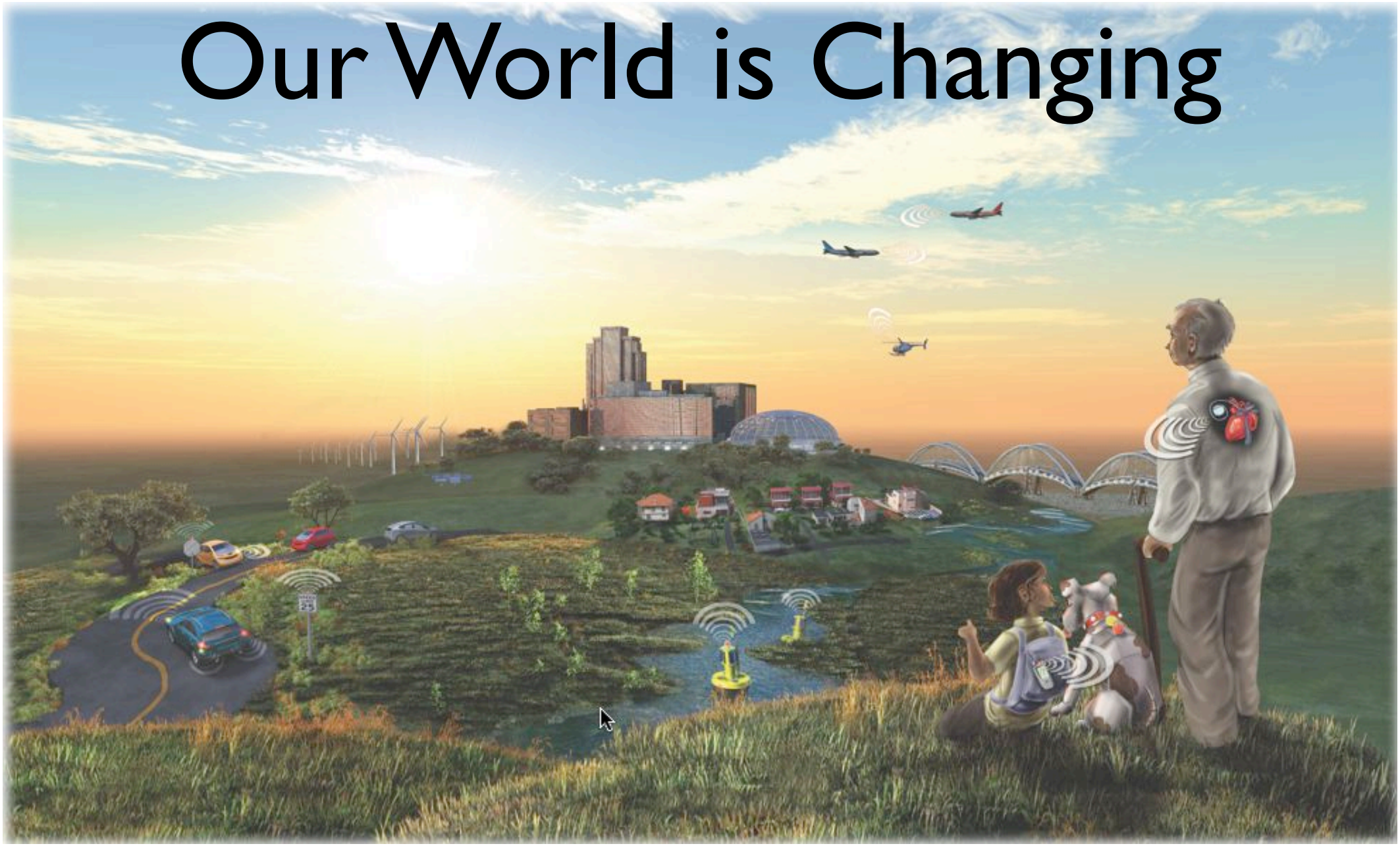
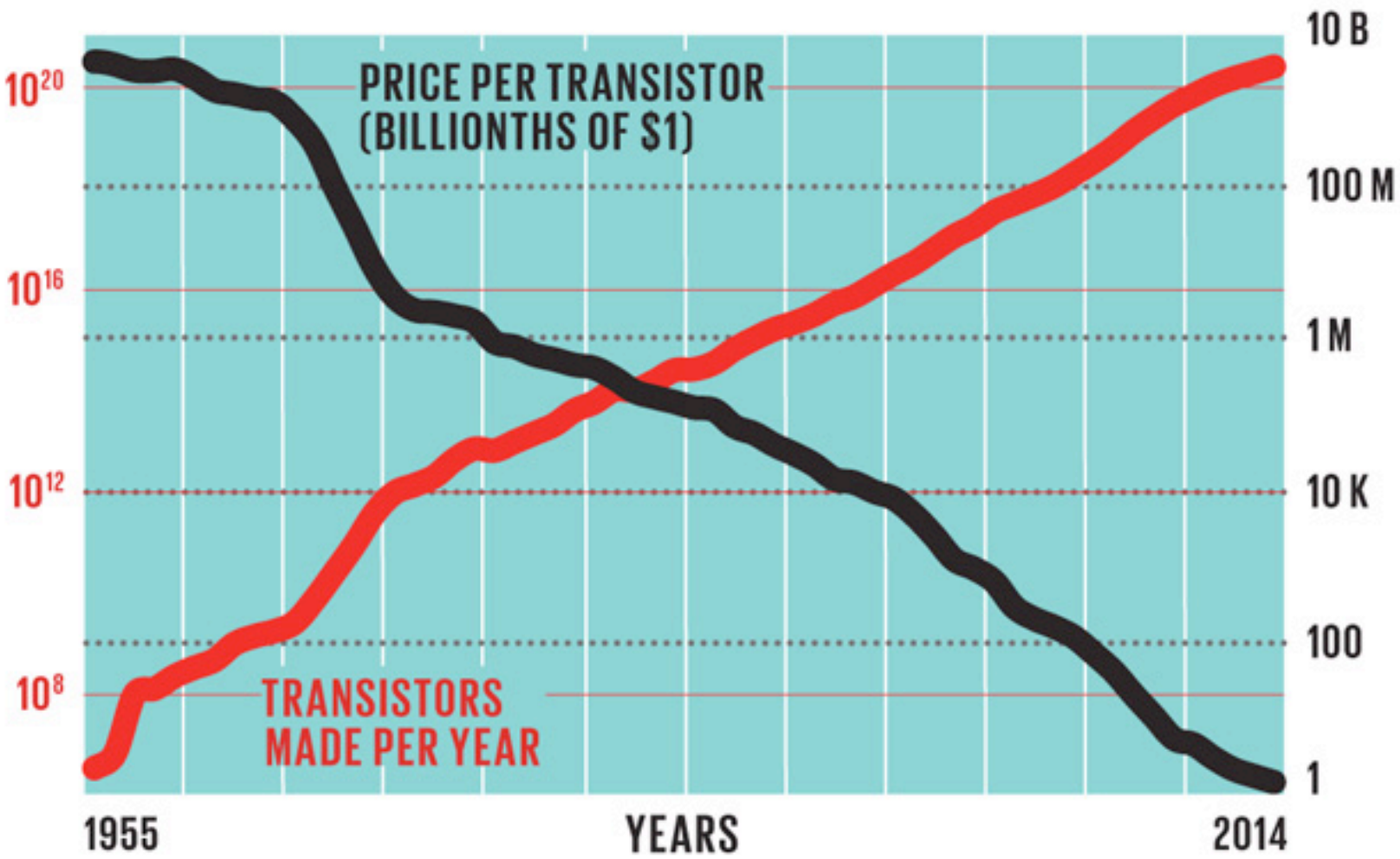
Rigorous Simulation

Walid Taha - Halmstad University

Our World is Changing

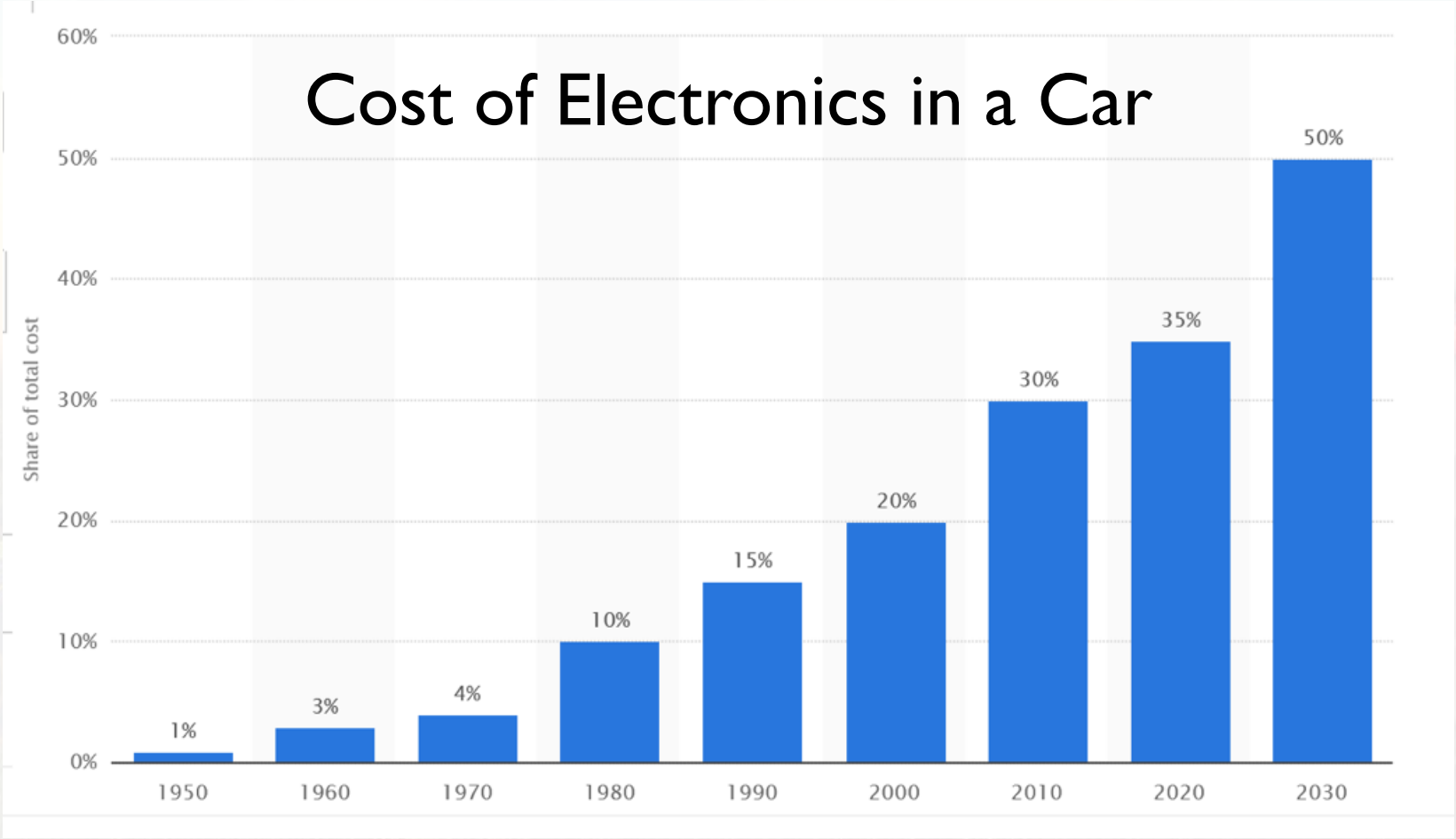


Our World is Changing



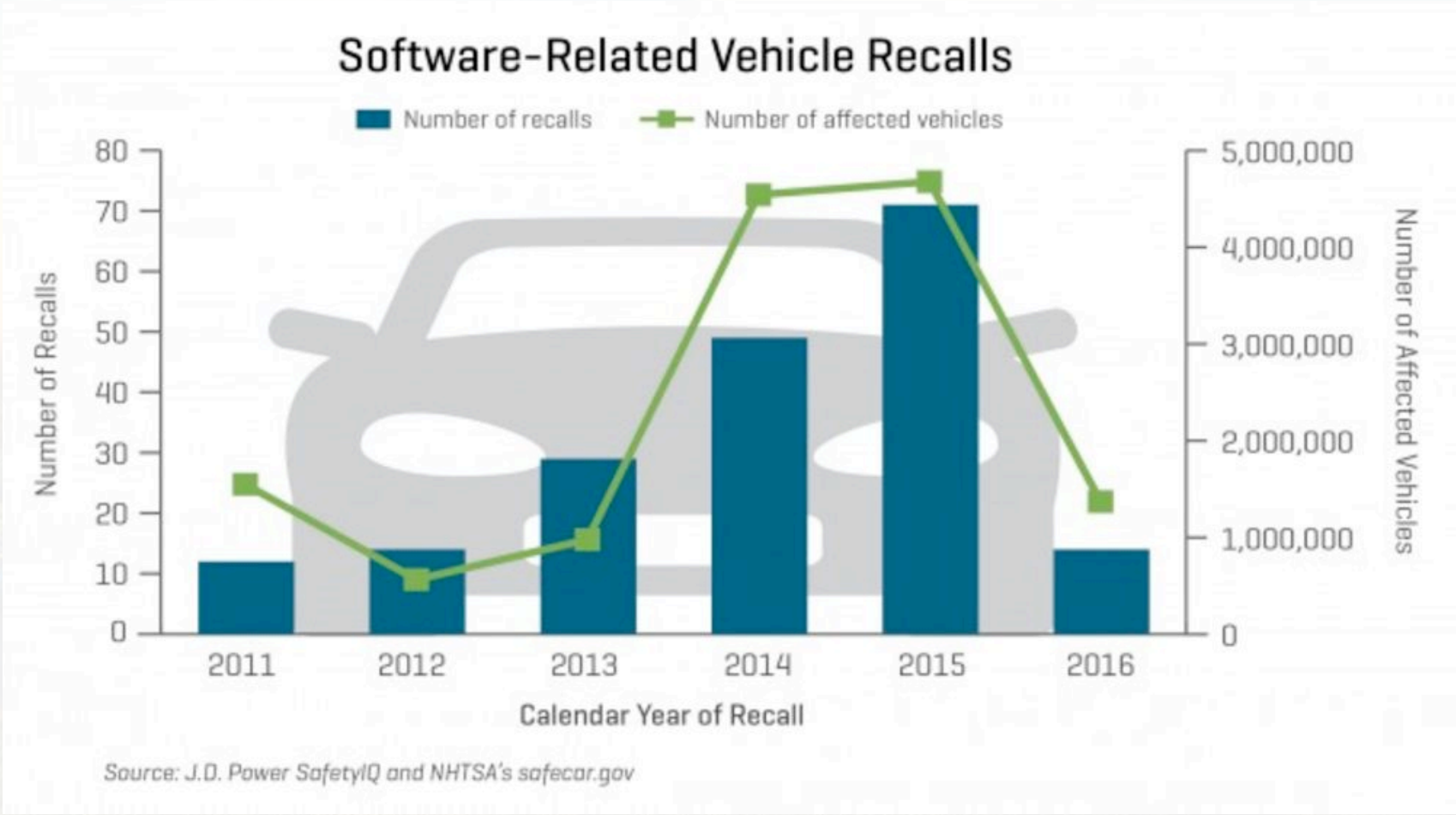
http://spectrum.ieee.org/computing/hardware/transistor-production-has-reached-astronomical-scales

Our World is Changing



http://electronicscomponentsworld.com/test-implications-arising-from-the-advent-of-connected-cars/

Our World is Changing



http://www.popsci.com/software-rising-cause-car-recalls

Problem

- Broadly: design technology is fragmented:
  - Software engineering environments
  - Simulation/Co-simulation
  - Symbolic algebra methods
  - Formal methods
- Specifically: formally correct simulation

Our Approach

"… if thought corrupts language, language can also corrupt thought."

George Orwell, *Politics and the English Language*, 1946 .

Domain-Specific Languages

- Our approach to DSL development:
  - Formalize the **notation** & **semantics**
  - **Implement**, **test**, **evaluate**, and
  - ... repeat
- 1st iteration 2006-2010
- 2nd iteration 2010-2016 - Today's focus

Notation & Expressivity

First Iteration Concept

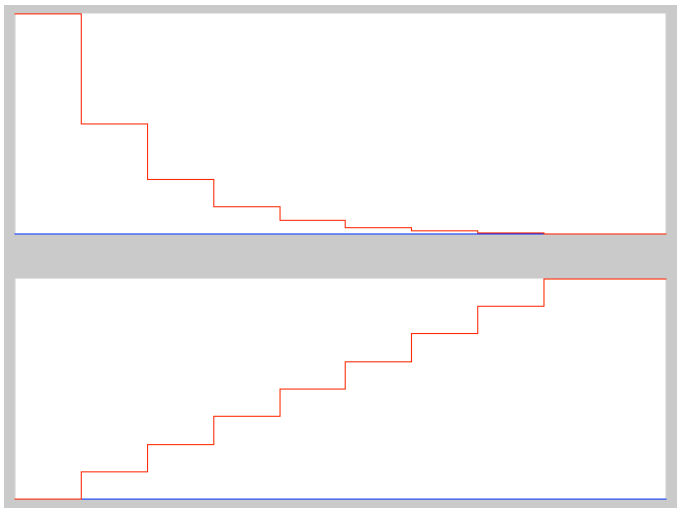
- Two different languages
  - One for embedded controllers (P-FRP)
    - Discrete events & strictly terminating
  - One for physical world outside
    - Time derivatives, partial derivatives, and undirected equations.

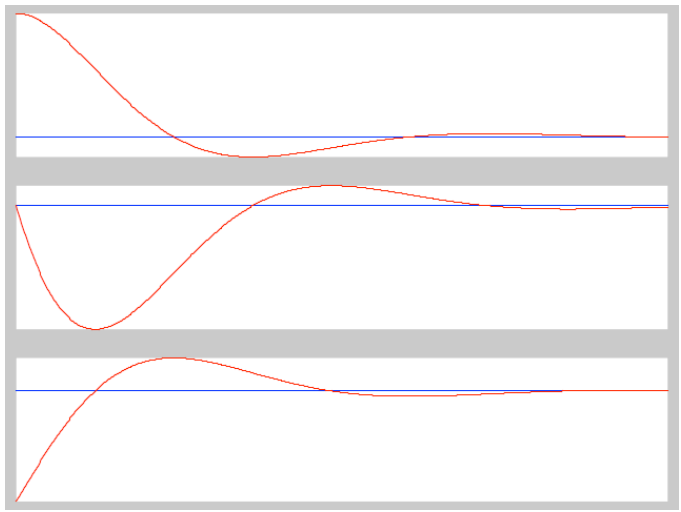
1st Iteration Concept

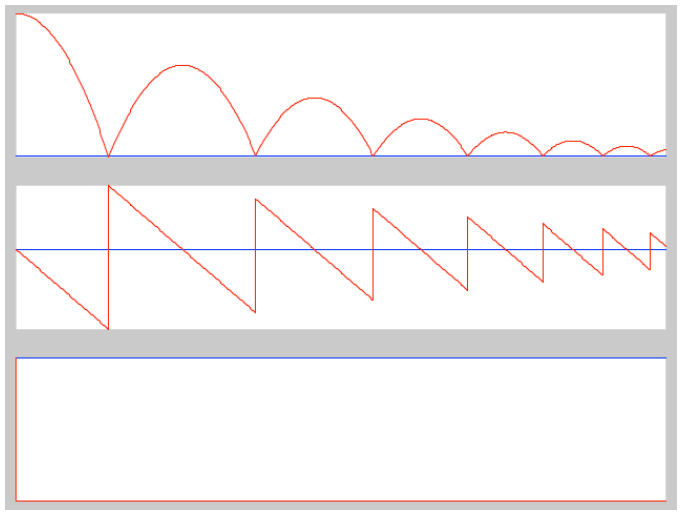
- After design, several interactions with
  - practitioners
  - other researchers
- Emerging questions:
  - “Why this particular controller language?”
  - “What if we want a continuous controller?”

Back to the Drawing Board

Discrete+Continuous=Hybrid







```
if x=>2 then
  x+ = x/2,
  n+ = n+1
noelse
```

$$x'' = -x - x'$$

```
if (x<0 && x'<0) then
  x'+ = -0.8*x'
else
  x"= -10
```

Minimal Core Calculi

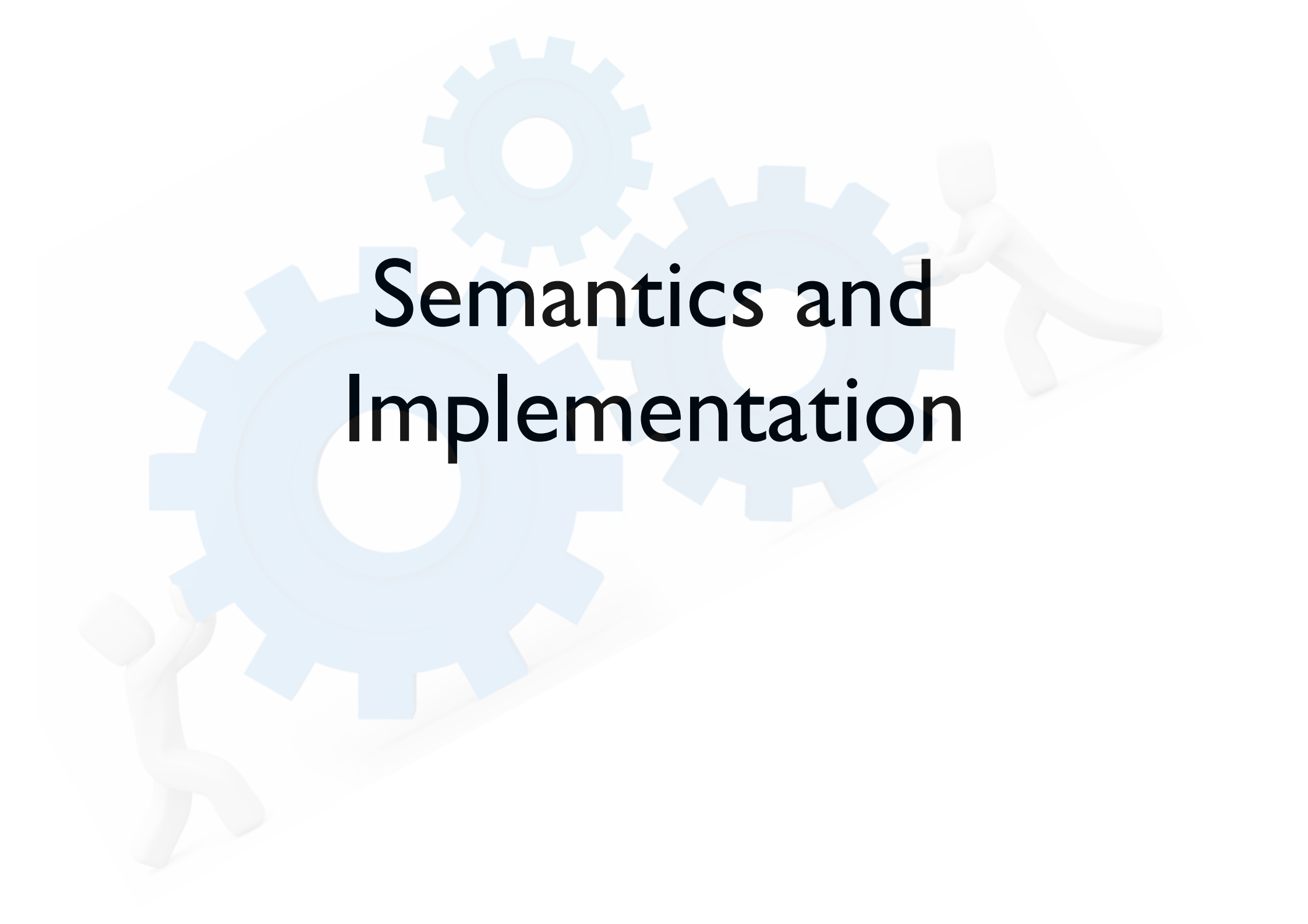
- MicroAcumen Disjoint guards, no nesting
  - Deno. Sem. (Moggi, Taha, Bartha)
  - Oper. Sem. (Duracz, Taha, Moggi, Bartha)
- MiniAcumen Adds nesting of conditionals
  - Quadratically shorter (Duracz, Taha)
- More on technical details later

Conservative Extensions

- Binding Time Analysis (BTA) and Automatic Differentiation enable two key extensions:
  - Equational constraints
  - Partial derivatives
- The combination is highly expressive
- Key achievement: Significant static checking and modular user feedback (Zeng, Taha)

Expressivity by Example

- Continuous plant, discrete controller
- A mechanical cam
- Exoskeleton arms (w O'Malley)
- Mechanics of bipedal robots (w Ames)
- Bouncing/sliding ball on a continuously varying surface



Semantics and Implementation

Symptoms and Problems

- Symptoms pointing to semantics questions
  - MATLAB/Simulink goes into infinite loop
  - Modelica semantics impl. dependent
- Problems
  - Hybrid systems introduce new pathologies
  - Unlike software and hardware, simulation cannot even be used for falsification

Pieces of the Puzzle

- Validated Numerics
  - Techniques for computing results with rigorous (guaranteed) bounds
  - Available today mainly as "solver" libraries
- Reachability Analysis
  - Most use symbolics, some val. numerics
  - Don’t deal well with pathologies

Hybrid Systems Pathologies

- Two main types:
  - Zeno: Infinite events in “finite” time
  - Chattering: Infinite events in zero time
- Cause simulators to either loop infinitely or produce incorrect results
- Cause reachability tools AND semantics to produce incorrect results

Rigorous Simulation of Zeno systems

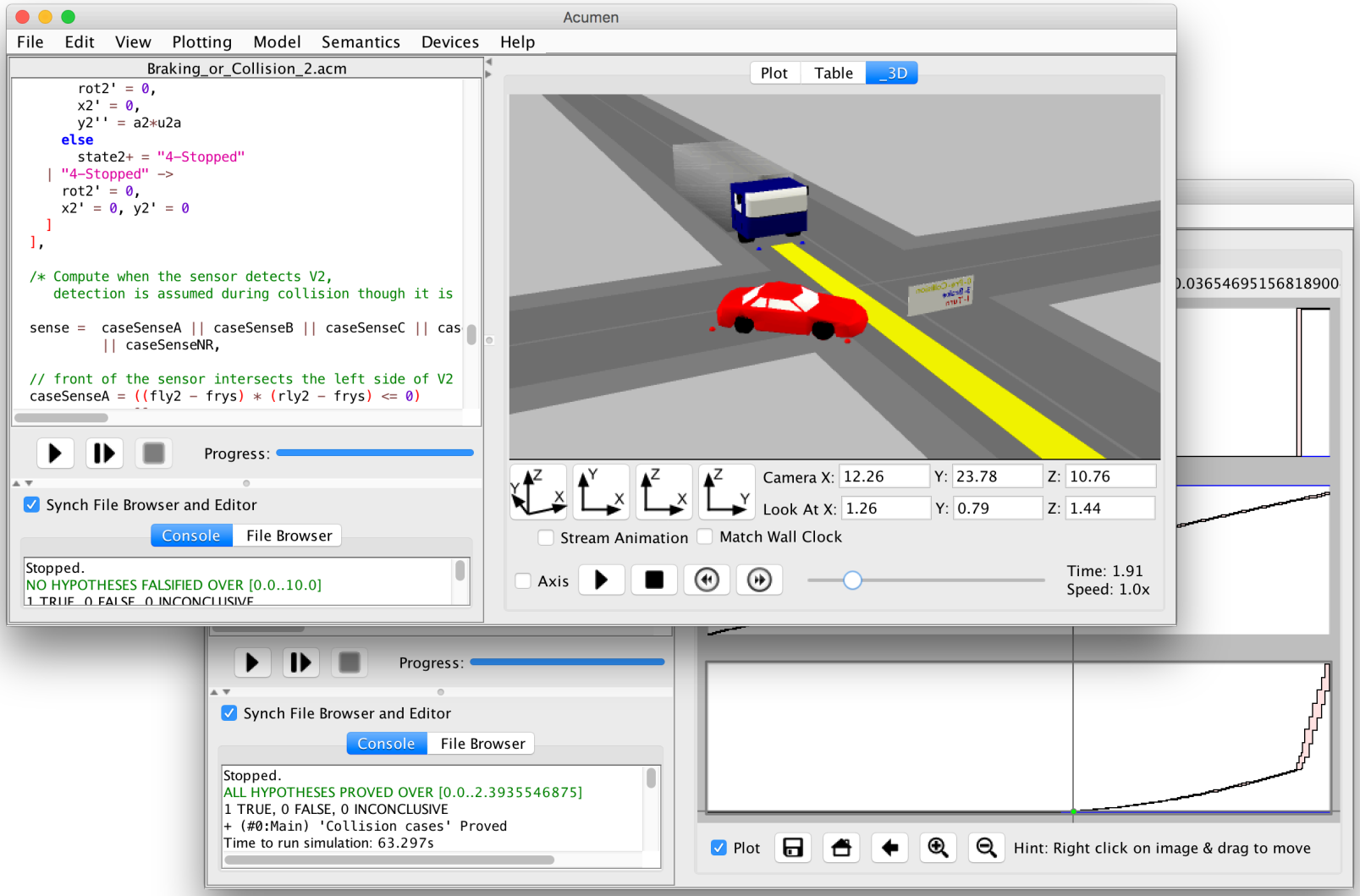
- Root cause of problem:
    - Simulators AND semantics use "events" as simulation "steps”
- Insight: These pathologies involve infinite events, but they are often "inconsequential"
- Validated methods are about sets, which facilitates checking fixed points (Konecny, Taha, Bartha, Duracz, Duracz, Ames)

Evaluating Emerging Design

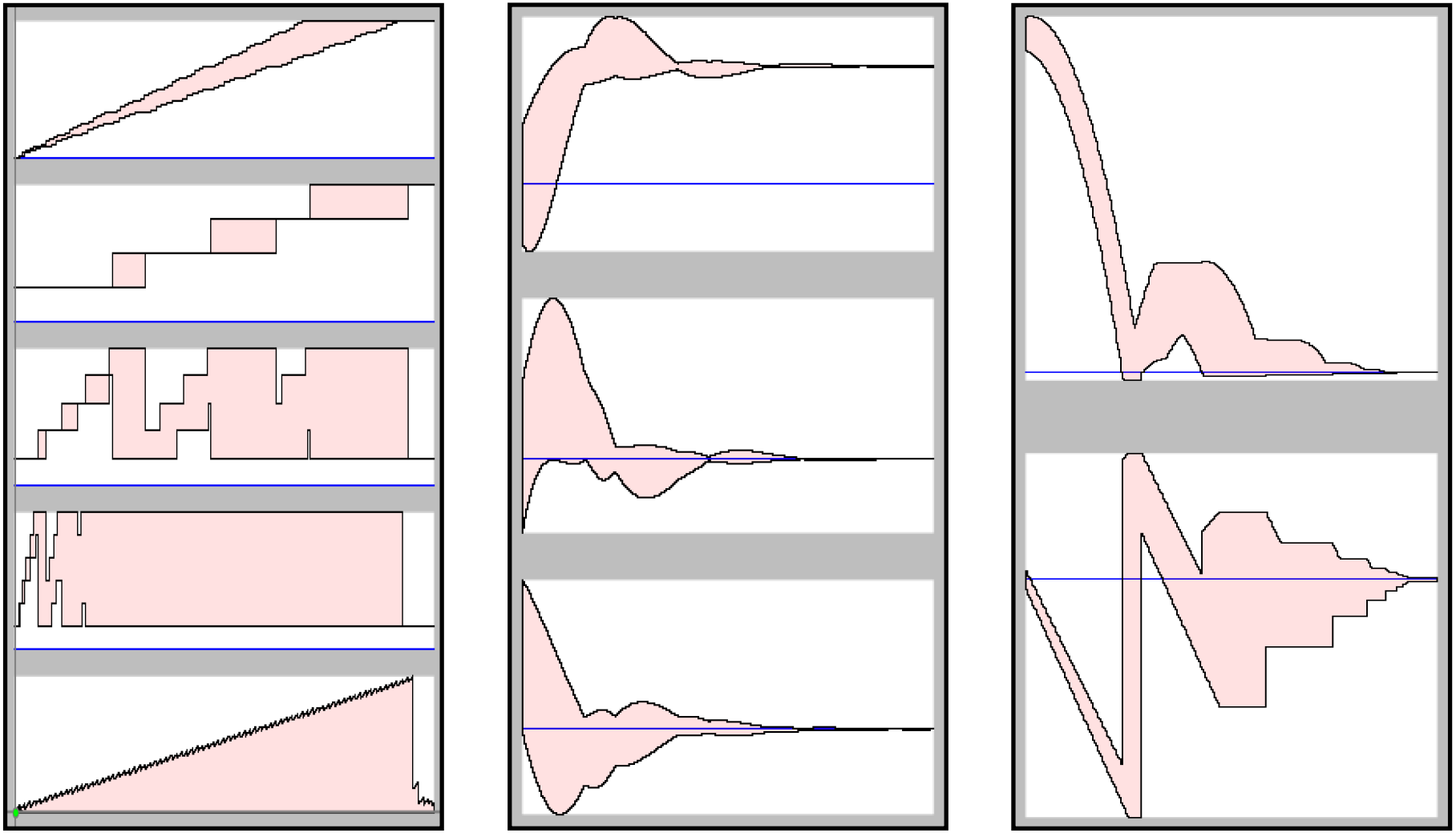
Second Iteration

- Sources of user feedback and observation
  - Used in a course on CPS for five years
  - Early domain expert feedback
  - Case study on advanced driving functions
  - Benchmarking support for robust design
- We say a bit more about the last two

Automotive Case Study



Support for Robust Design



Ongoing & Future Work

- Key areas for for third iteration are:
  - Soundness of op./deno. Semantics
    - & of core validated numerics algorithms
  - Minimization (of scalars and functions)
  - Rigorous bounds on prob. distributions

Publications & Acumen

- All papers available online
  - www.effective-modeling.org
- Acumen open source code:
  - www.acumen-language.org
- Most recent dissertation from group:
  - bit.ly/adam-thesis

Conclusions

- Rigorous simulation
  - addresses foundational problems
  - provides fully automated proofs
    - about behavior of hybrid systems
    - which are useful models of many CPSs
- Open source implementation (Acumen)

All related publications can be found at http://www.effective-modeling.org/p/publications.html