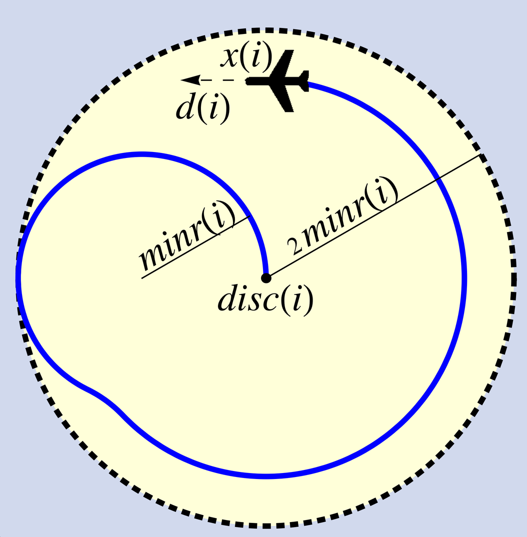
KeYmaera and Sφnx for Hybrid System Modeling and Verification

Stefan Mitsch, Sarah M. Loos, and André Platzer
Computer Science Department, Carnegie Mellon University

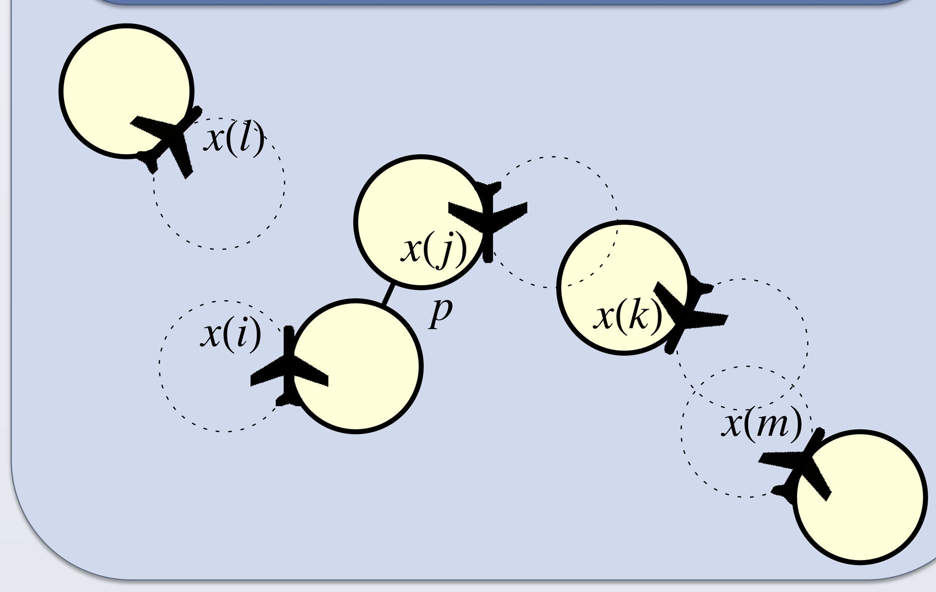
CASE STUDIES

Aircraft Controllers [1]: As airspace becomes ever more crowded, air traffic management must reduce both space and time between aircraft to increase throughput, making on-board collision avoidance systems ever more important. We prove the collision avoidance systems never allow aircraft to get too close to one another, even when new planes approach an in-progress avoidance maneuver.

Aircraft Big Disc Protocol

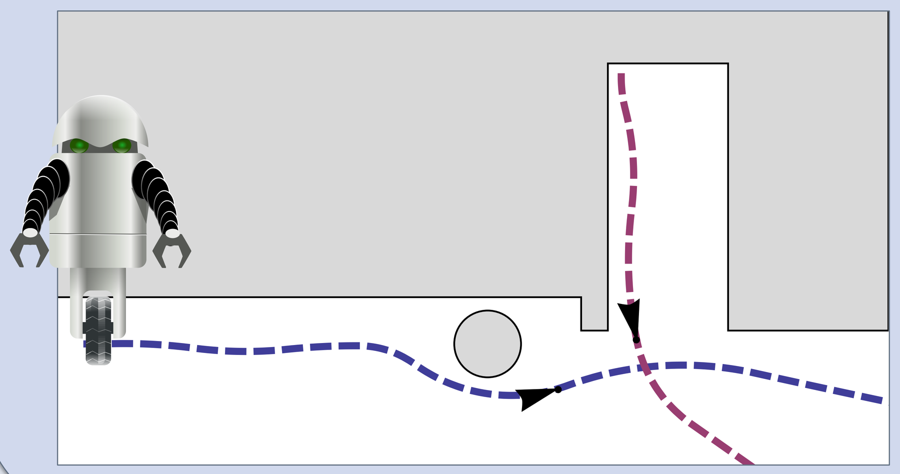


Aircraft Small Discs Protocol



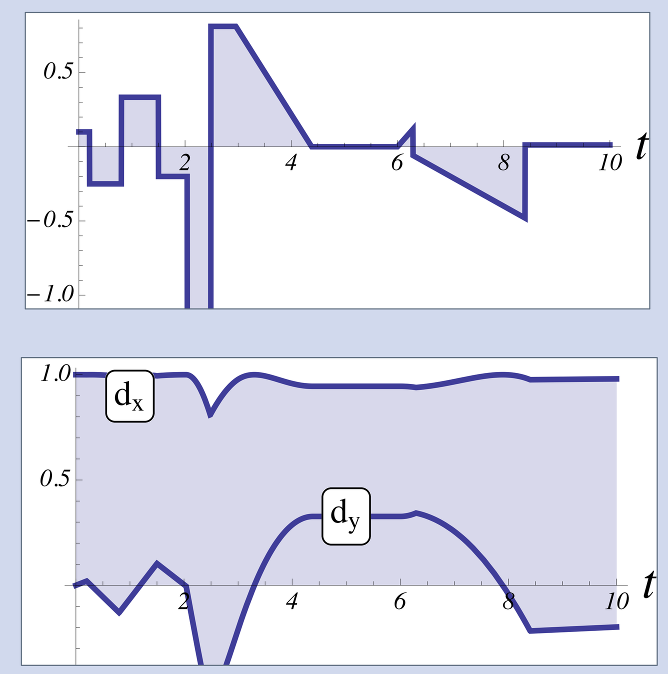
Robot Obstacle Avoidance [2]: Nowadays, robots interact frequently with a dynamic environment outside limited manufacturing sites and in close proximity with humans. Thus, safety of motion and obstacle avoidance are vital safety features of such robots. We formally verify that our controller avoids both stationary and moving obstacles.

Robot Obstacle Avoidance: Passive Safety



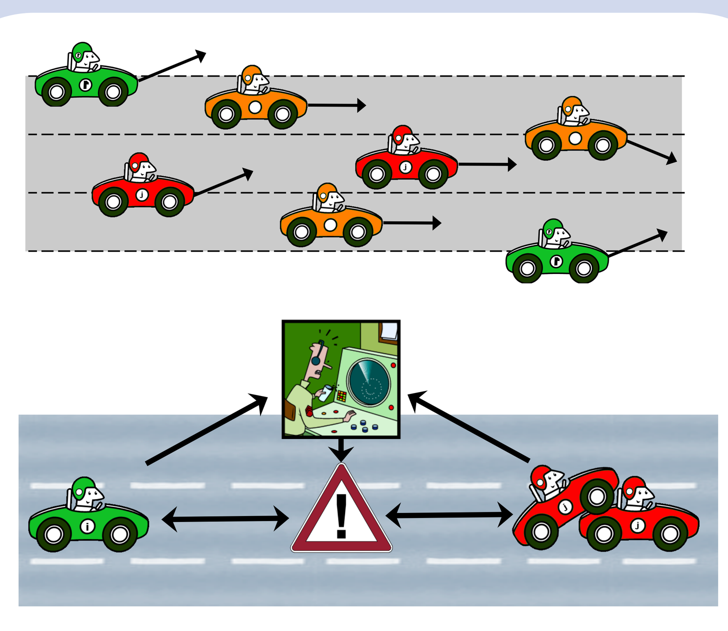
Train Control [3]: We prove that the European Train Control System protocol remains correct even in the presence of perturbation by disturbances in the dynamics and when a PI controlled speed supervision is used.

Hybrid Dynamics



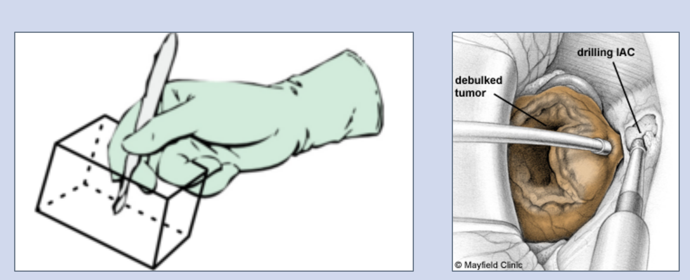
Adaptive Cruise Control, Road Traffic Control [4,5]: Car safety measures can be most effective when the cars on a street coordinate their control actions to minimize the risk of safety hazards and collisions. We verify that the control models guarantee collision freedom for arbitrarily many cars on a street, even if new cars enter from on-ramps or multi-lane streets.

Adaptive Cruise Control
and Road Traffic Control



Robotic Surgery [6]: We analyzed a control algorithm that provides directional force feedback for a surgical robot. We found two bugs and described exactly what could go wrong. We then developed a new algorithm that provides safe operation along with directional force feedback. We created a machine-checked proof that guarantees the new algorithm is safe for all inputs.

Robotic Surgery



METHOD

Automotive, aircraft, railway, and robotics controllers are hybrid systems, which we model and verify using differential dynamic logic (dL) [7,8]. Below are a few selected rules from the dL proof calculus.

$$\frac{\phi_x^{f(x)}}{[x := f(x)]\phi}$$

$$\frac{\exists t{\geq}0\,\langle x := y_x(t)\rangle\phi}{\langle x' = f(x)\rangle\phi}$$

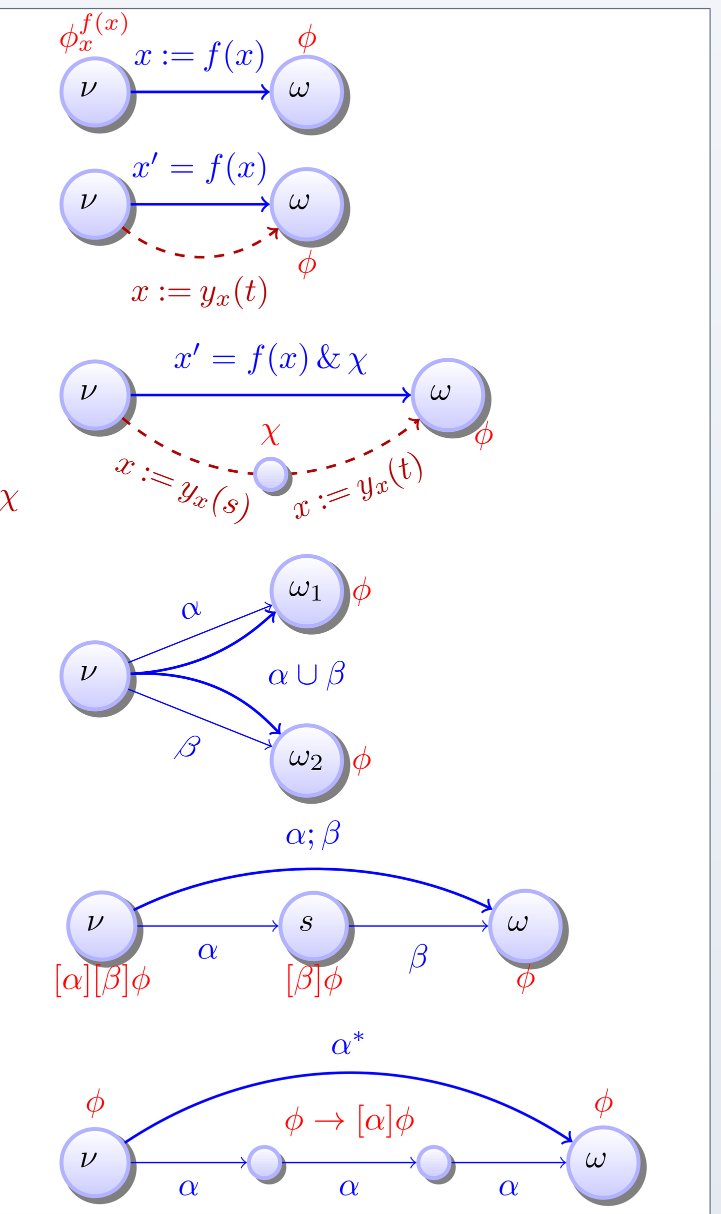
$$\frac{\exists t{\geq}0\,(\bar\chi \land \langle x := y_x(t)\rangle\phi)}{\langle x' = f(x)\,\&\,\chi\rangle\phi}$$

$$\bar\chi \equiv \forall 0{\leq}s{\leq}t\,\langle x := y_x(s)\rangle\chi$$

$$\frac{[\alpha]\phi \land [\beta]\phi}{[\alpha\cup\beta]\phi}$$

$$\frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi}$$

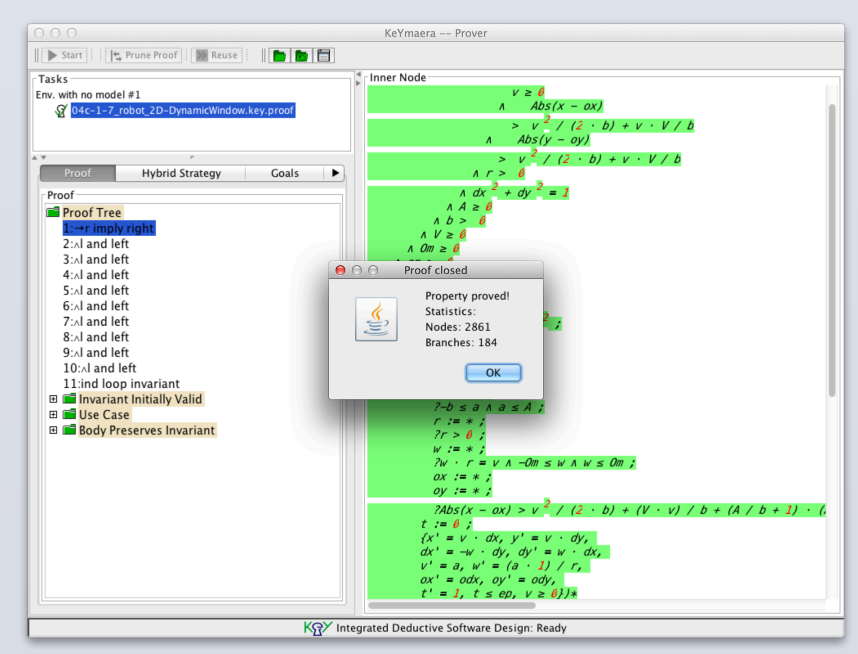
$$\frac{\vdash \forall^\alpha(\phi \to [\alpha]\phi)}{\phi \vdash [\alpha^*]\phi}$$



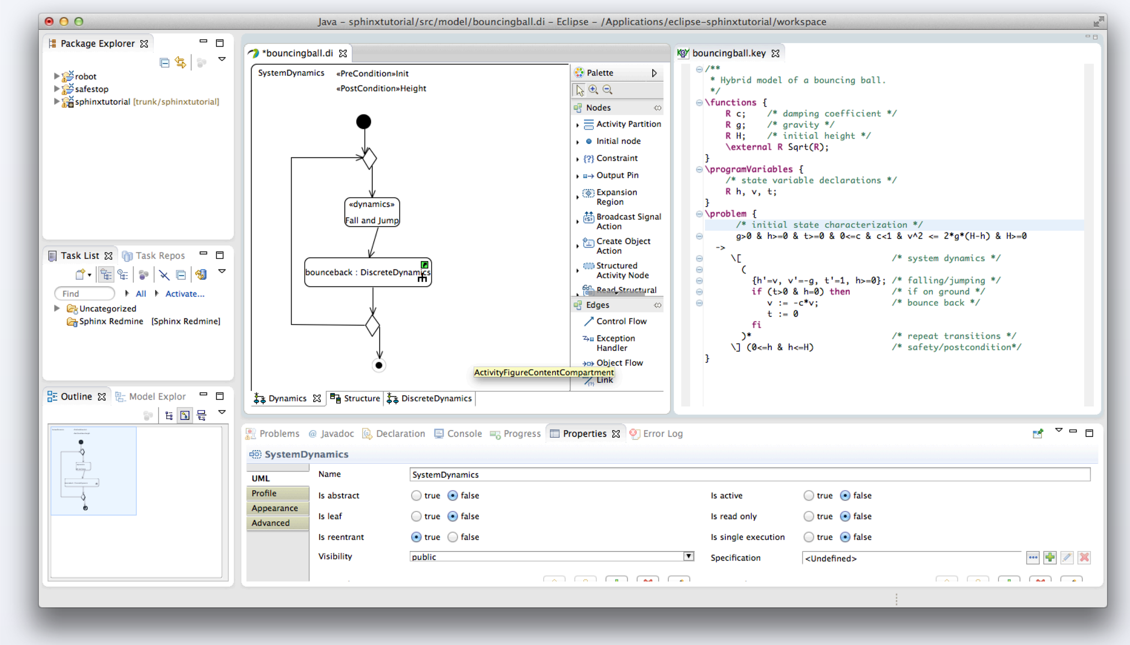
The basic operators are (non-deterministic) assignment (x := f(x), x := *), non-deterministic choice (α U β), sequential composition (α;β), and non-deterministic repetition (α*). dL formulae are built with expressions over the reals using the usual logical connectives, quantifiers, and modality operators.

TOOLS

KeYmaera [9] is our automated theorem-prover for differential dynamic logic. KeYmaera supports hybrid systems with nonlinear discrete jumps, non-linear differential equations, differential-algebraic equations, differential inequalities, and systems with nondeterministic discrete or continuous input. KeYmaera implements proof strategies that decompose hybrid systems symbolically and prove the full system by proving properties of its parts.



Sφnx [10] is our model-based design tool for hybrid systems. It provides a graphical and textual editor for differential dynamic logic. Sφnx links to KeYmaera, exchanges models and full or partial proofs, and supports C code generation.



HOW CAN I APPLY THE METHODS?

Our formal verification methods use a hybrid system model that specifies both its discrete and continuous behavior. This tends to be easier if the original specification already models physics (e.g., differential equations for robot kinematics). In addition, the safety condition(s) that must be met by the system must be formally specified. Sometimes, these are obvious (e.g., distance to static obstacles always non-zero), in other cases the process of finding a safety condition already provides value in itself (e.g., who is to blame for a collision of two moving agents?).

REFERENCES

[1] Sarah M. Loos, David W. Renshaw and André Platzer. Formal Verification of Distributed Aircraft Controllers. *In Calin Belta and Franjo Ivancic, editors, HSCC 2013.*

[2] Stefan Mitsch, Khalil Ghorbal and André Platzer. On Provably Safe Obstacle Avoidance for Autonomus Robotic Ground Vehicles. *In RSS 2013.*

[3] André Platzer and Jan-David Quesel. European Train Control System: A case study in formal verification. *In Karin Breitman and Ana Cavalcanti, editors, ICFEM 2009.*

[4] Sarah M. Loos, André Platzer and Ligia Nistor. Adaptive cruise control: Hybrid, distributed, and now formally verified. *In Michael Butler and Wolfram Schulte, editors, FM 2011.*

[5] Stefan Mitsch, Sarah M. Loos and André Platzer. Towards formal verification of freeway traffic control. *In Chenyang Lu, ICCPS 2012.*

[6] Yanni Kouskoulas, David W. Renshaw, André Platzer and Peter Kazanzides. Certifying the safe design of a virtual fixture control algorithm for a surgical robot. *In Calin Belta and Franjo Ivancic, editors, HSCC 2013.*

[7] André Platzer. Differential dynamic logic. J. Autom. Reasoning 41(2), 2008.

[8] André Platzer. Logics of dynamical systems. *In ACM/IEEE Symposium on Logic in Computer Science, LICS 2012.*

[9] André Platzer and Jan-David Quesel. KeYmaera: A hybrid theorem prover for hybrid systems. *In Alessandro Armando, Peter Baumgartner and Gilles Dowek, IJCAR 2008.*

[10] Stefan Mitsch, Grant O. Passmore and André Platzer. A Vision of Collaborative Verification-Driven Engineering of Hybrid Systems. *In Manfred Kerber, Christoph Lange and Colin Rowat, Do-Form 2013.*