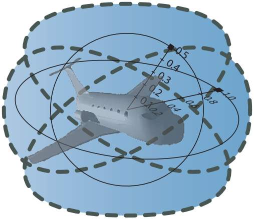
Logic & Proofs for Cyber-Physical Systems

André Platzer

aplatzer@cs.cmu.edu
Computer Science Department
Carnegie Mellon University, Pittsburgh, PA



Outline

1. CPS are Multi-Dynamical Systems
   - Hybrid Systems
   - Hybrid Games
   - Stochastic Hybrid Systems
   - Distributed Hybrid Systems
2. Dynamic Logic of Multi-Dynamical Systems
3. Proofs for CPS
4. Theory of CPS
   - Soundness and Completeness
   - Differential Invariants
   - Differential Axioms
   - Example: Elementary Differential Invariants
5. Applications
6. Summary

Cyber-Physical Systems Analysis: Aircraft Example

Which control decisions are safe for aircraft collision avoidance?

Cyber-Physical Systems

CPSs combine cyber capabilities with physical capabilities
to solve problems that neither part could solve alone.

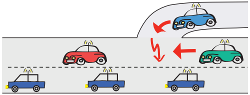
CPSs Promise Transformative Impact!

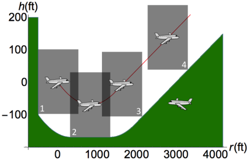
Prospects: Safe & Efficient

Driver assistance
Autonomous cars

Pilot decision support
Autopilots / UAVs

Train protection
Robots near humans





Prerequisite: CPSs need to be safe

How do we make sure CPSs make the world a better place?

Can you trust a computer to control physics?

Can you trust a computer to control physics?

1. Depends on how it has been programmed
2. And on what will happen if it malfunctions

Rationale

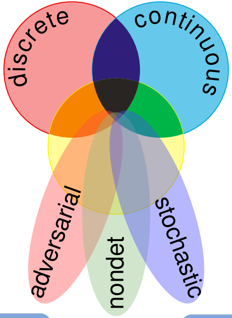
1. Safety guarantees require analytic foundations.
2. A common foundational core helps all application domains.
3. Foundations revolutionized digital computer science & our society.
4. Need even stronger foundations when software reaches out into our physical world.

CPSs deserve proofs as safety evidence!

CPSs are Multi-Dynamical Systems

CPS Dynamics

CPS are characterized by multiple facets of dynamical systems.



CPS Compositions

CPS combines multiple simple dynamical effects.

Descriptive simplification

Tame Parts

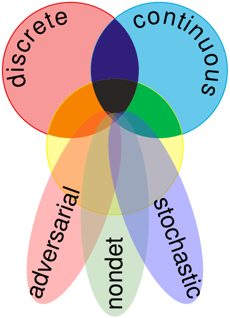
Exploiting compositionality tames CPS complexity.

Analytic simplification

CPSs are Multi-Dynamical Systems

hybrid systems

HS = discrete + ODE

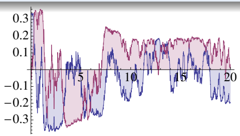


hybrid games

HG = HS + adversary

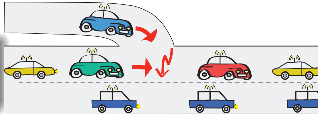
stochastic hybrid sys.

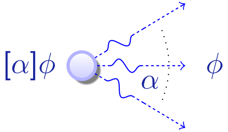
SHS = HS + stochastics



distributed hybrid sys.

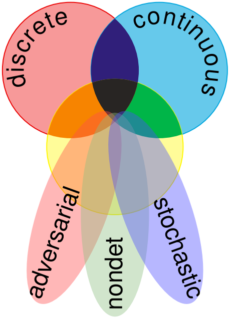
DHS = HS + distributed





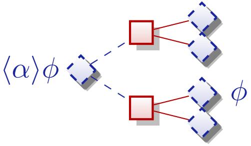
differential dynamic logic

$d\mathcal{L} = \mathrm{DL} + \mathrm{HP}$



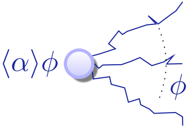
differential game logic

$dG\mathcal{L} = \mathrm{GL} + \mathrm{HG}$



stochastic differential DL

$Sd\mathcal{L} = \mathrm{DL} + \mathrm{SHP}$



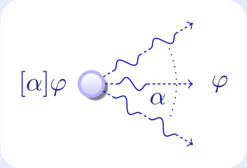
quantified differential DL

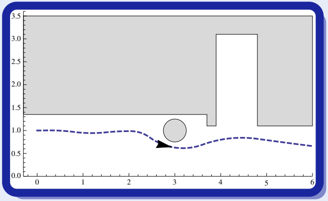
$Qd\mathcal{L} = \mathrm{FOL} + \mathrm{DL} + \mathrm{QHP}$

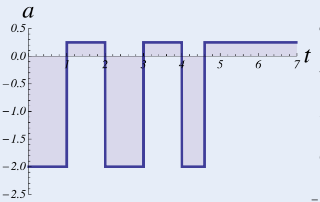
JAR'08,CADE'11,LMCS'12,LICS'12

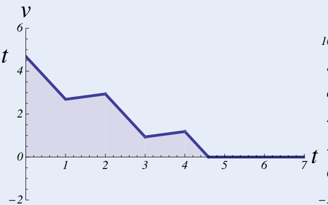
LICS'12,CADE'15,TOCL'15

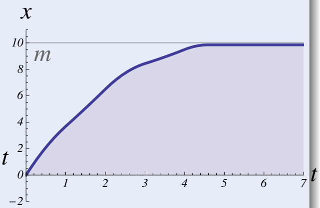
Concept (Differential Dynamic Logic) (JAR'08,LICS'12)





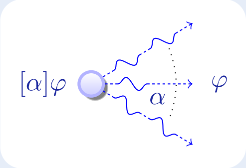


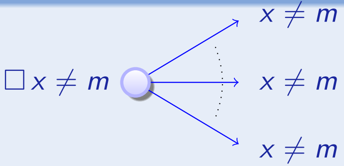


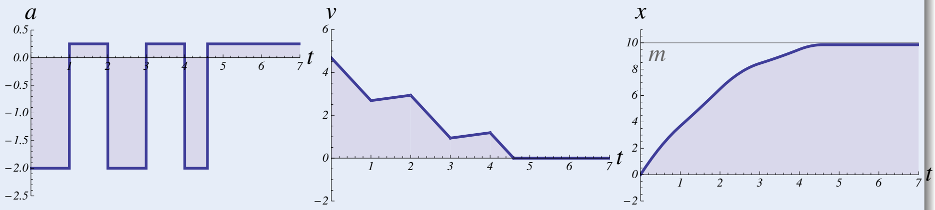


CPS Analysis

Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

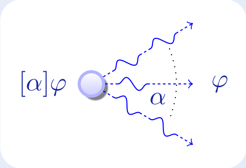


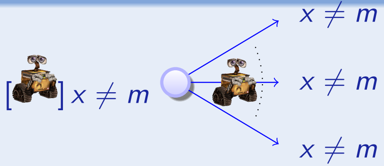


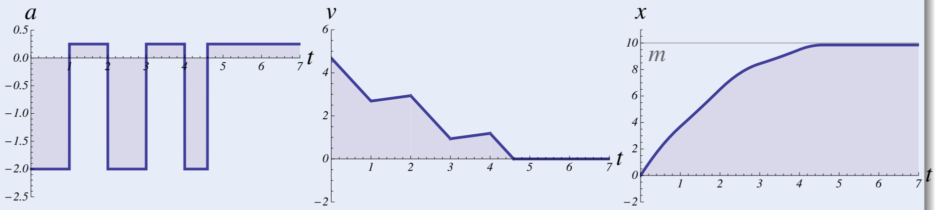


CPS Analysis

Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

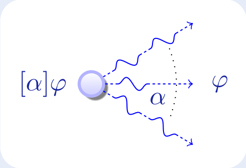


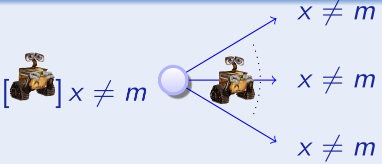




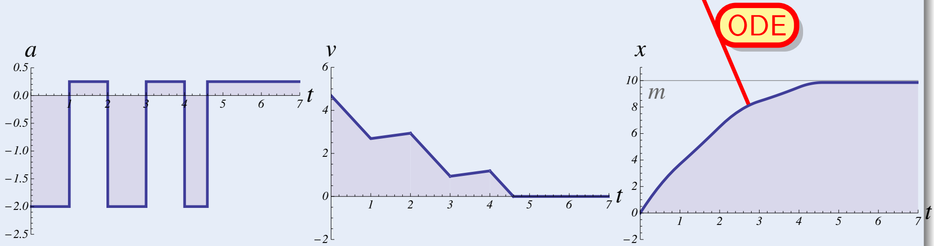
CPS Analysis

Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

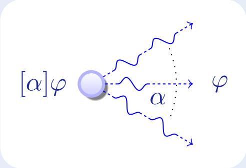


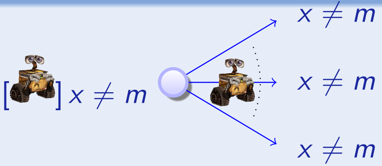


$x' = v, v' = a$

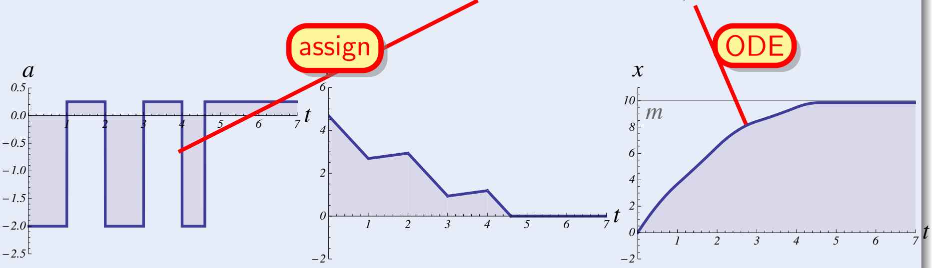


Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

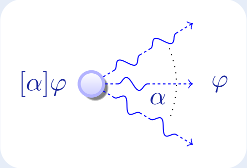


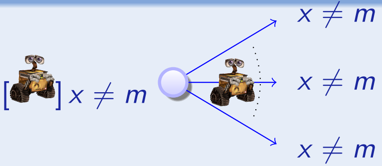


$a := -b$ $\quad x' = v, v' = a$

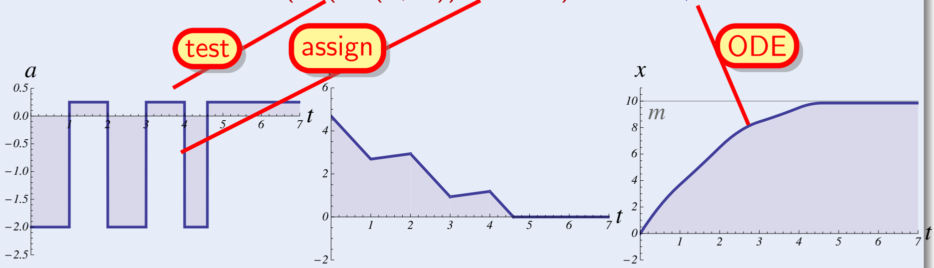


Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

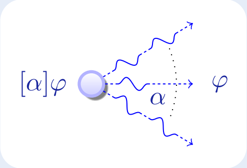




$(\texttt{if}(\mathrm{SB}(x, m))\ a := -b) \quad x' = v, v' = a$



Concept (Differential Dynamic Logic) (JAR'08,LICS'12)



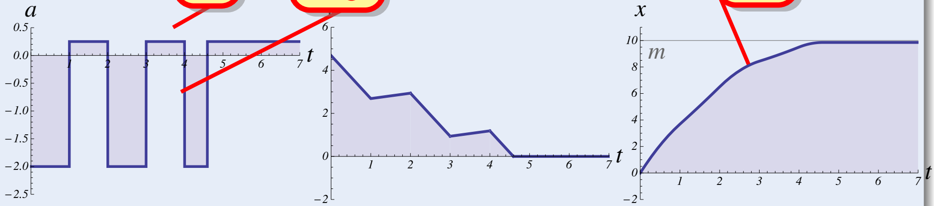
seq. compose

$(\texttt{if}(SB(x,m))\ a := -b)\ ;\ x' = v, v' = a$

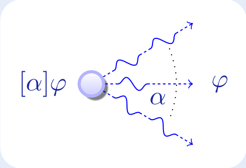
test

assign

ODE



Concept (Differential Dynamic Logic) (JAR'08,LICS'12)



$$\left(\left(\mathtt{if}(SB(x,m))\ a := -b\right)\ ;\ x' = v, v' = a\right)^*$$

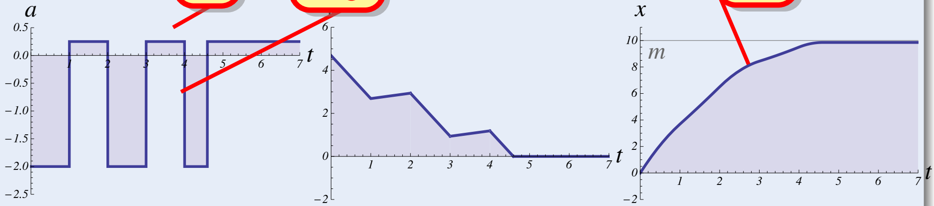
seq. compose

nondet. repeat

test

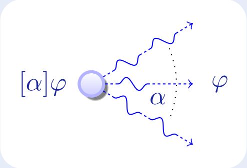
assign

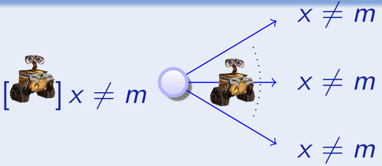
ODE



CPS Analysis

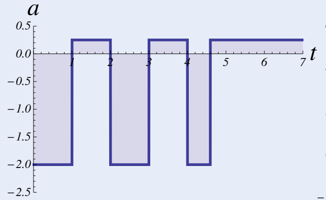
Concept (Differential Dynamic Logic) (JAR'08,LICS'12)

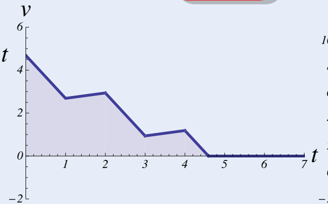


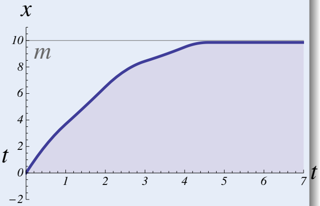


$$\left[\left(\left(\mathtt{if}(SB(x,m))\ a := -b\right)\ ;\ x' = v, v' = a\right)^*\right] \underbrace{x \neq m}_{\text{post}}$$

all runs

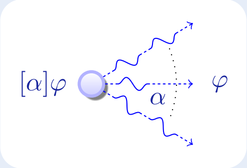


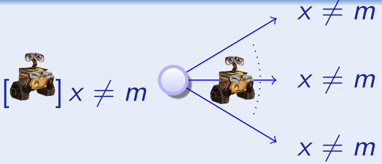




CPS Analysis

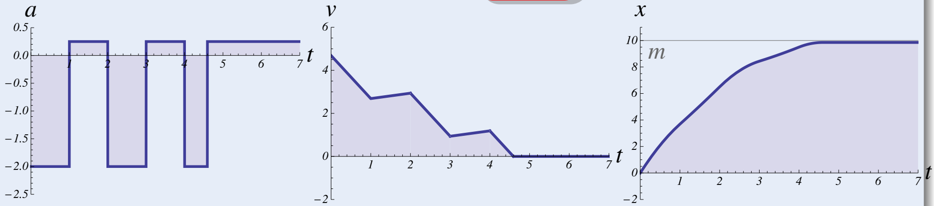
Concept (Differential Dynamic Logic) (JAR'08,LICS'12)





$$\underbrace{x \neq m \land b > 0}_{\text{init}} \rightarrow \big[\big((\mathtt{if}(SB(x, m))\ a := -b)\ ;\ x' = v, v' = a\big)^*\big]\underbrace{x \neq m}_{\text{post}}$$

all runs



Differential Dynamic Logic d$\mathcal{L}$: Syntax

Definition (Hybrid program α)

$$x := f(x) \mid ?Q \mid x' = f(x) \,\&\, Q \mid \alpha \cup \beta \mid \alpha;\beta \mid \alpha^*$$

Definition (d$\mathcal{L}$ Formula P)

$$e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid [\alpha]P \mid \langle\alpha\rangle P$$

Tableaux'07,JAutomReas'08,LICS'12

Differential Dynamic Logic d$\mathcal{L}$: Syntax

Discrete Assign | Test Condition | Differential Equation | Nondet. Choice | Seq. Compose | Nondet. Repeat

Definition (Hybrid program α)

$$x := f(x) \mid ?Q \mid x' = f(x) \,\&\, Q \mid \alpha \cup \beta \mid \alpha;\beta \mid \alpha^*$$

Definition (d$\mathcal{L}$ Formula P)

$$e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid [\alpha]P \mid \langle\alpha\rangle P$$

All Reals | Some Reals | All Runs | Some Runs

Tableaux'07,JAutomReas'08,LICS'12

Differential Dynamic Logic: Axiomatization

equations of truth

[:=]	$[x := e]P(x) \leftrightarrow P(e)$
[?]	$[?Q]P \leftrightarrow (Q \rightarrow P)$
[']	$[x' = f(x)]P \leftrightarrow \forall t{\geq}0\, [x := y(t)]P \qquad (y'(t) = f(y))$
[∪]	$[\alpha \cup \beta]P \leftrightarrow [\alpha]P \wedge [\beta]P$
[;]	$[\alpha; \beta]P \leftrightarrow [\alpha][\beta]P$
[*]	$[\alpha^*]P \leftrightarrow P \wedge [\alpha][\alpha^*]P$
K	$[\alpha](P \rightarrow Q) \rightarrow ([\alpha]P \rightarrow [\alpha]Q)$
I	$[\alpha^*](P \rightarrow [\alpha]P) \rightarrow (P \rightarrow [\alpha^*]P)$
C	$[\alpha^*]\forall v{>}0\, (P(v) \rightarrow \langle\alpha\rangle P(v-1)) \rightarrow \forall v\, (P(v) \rightarrow \langle\alpha^*\rangle \exists v{\leq}0\, P(v))$

LICS'12,CADE'15

Complete Proof Theory of Hybrid Systems

Theorem (Sound & Complete) (J.Autom.Reas. 2008, LICS'12)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to either differential equations* **or** *to discrete dynamics.* Proof 25pp

Corollary (Complete Proof-theoretical Bridge)

proving continuous = proving hybrid = proving discrete

JAutomReas'08,LICS'12

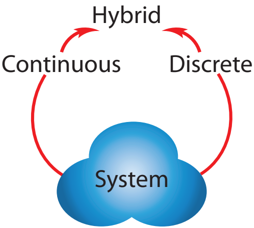
Complete Proof Theory of Hybrid Systems

Theorem (Sound & Complete) (J.Autom.Reas. 2008, LICS'12)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to either differential equations* **or** *to discrete dynamics.* ▸ Proof 25pp

Corollary (Complete Proof-theoretical Bridge)

proving continuous = proving hybrid = proving discrete



JAutomReas'08,LICS'12

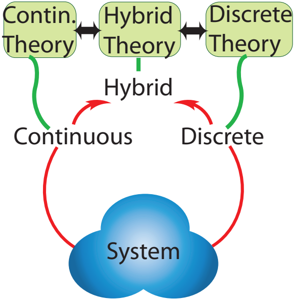
Complete Proof Theory of Hybrid Systems

Theorem (Sound & Complete) (J.Autom.Reas. 2008, LICS'12)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to either differential equations* **or** *to discrete dynamics.* ▸ Proof 25pp

Corollary (Complete Proof-theoretical Bridge)

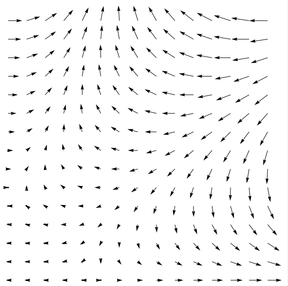
proving continuous = proving hybrid = proving discrete



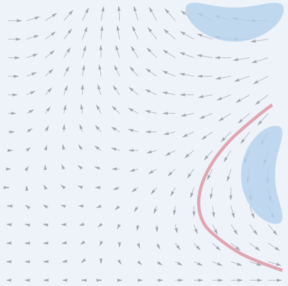
JAutomReas'08,LICS'12

Differential Invariants for Differential Equations

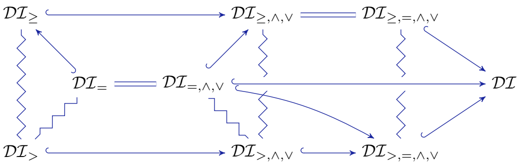
Differential Invariant



Differential Cut



Differential Ghost



Logic

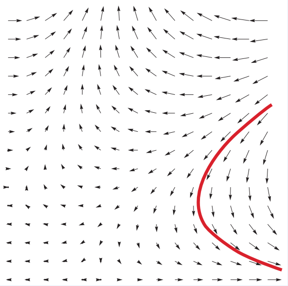
Provability theory

Math

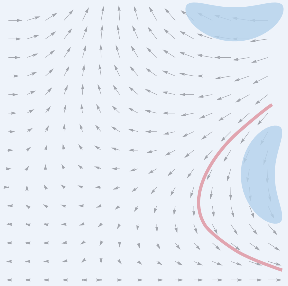
Character-istic PDE

JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

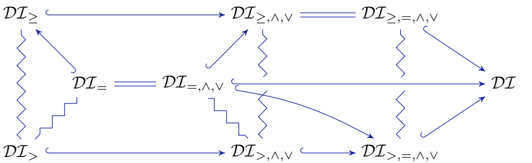
Differential Invariant



Differential Cut



Differential Ghost



Logic

Provability theory

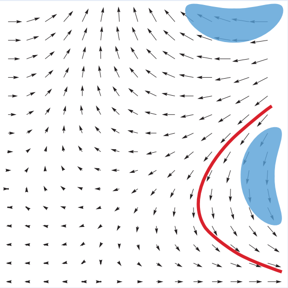
Math

Characteristic PDE

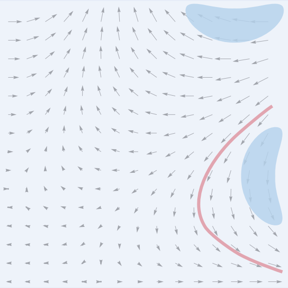
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

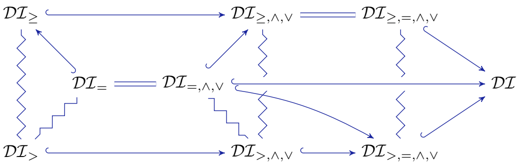
Differential Invariant



Differential Cut



Differential Ghost



Logic

Provability theory

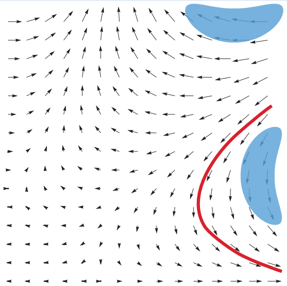
Math

Characteristic PDE

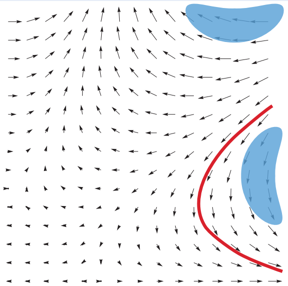
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

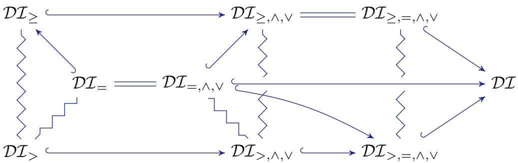
Differential Invariant



Differential Cut



Differential Ghost



Logic

Provability theory

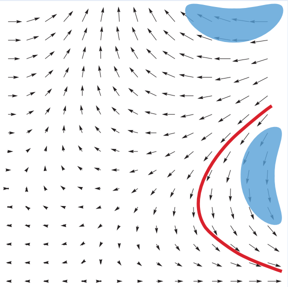
Math

Character-istic PDE

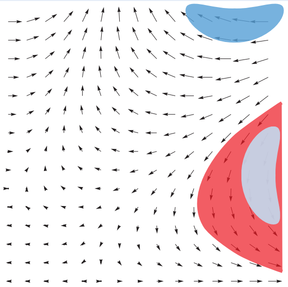
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

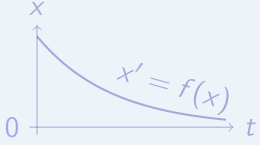
Differential Invariant

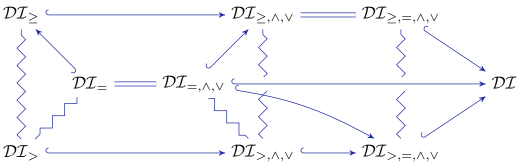


Differential Cut



Differential Ghost





Logic

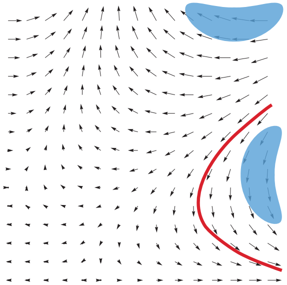
Provability theory

Math

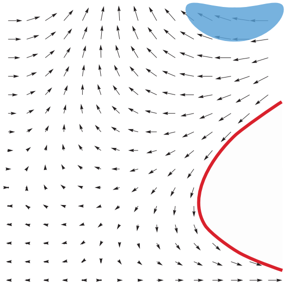
Character-istic PDE

JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

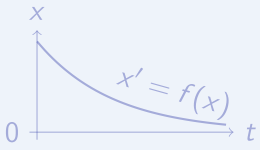
Differential Invariant

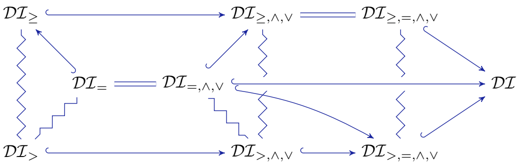


Differential Cut



Differential Ghost





Logic

Provability theory

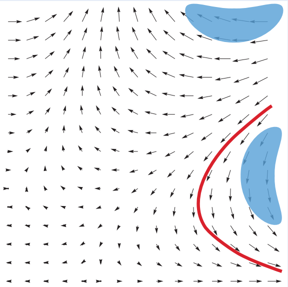
Math

Character-istic PDE

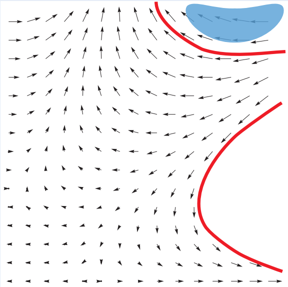
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

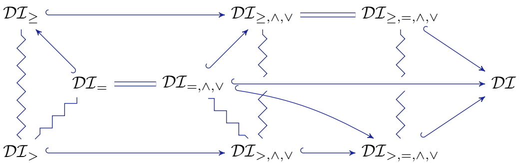
Differential Invariant



Differential Cut



Differential Ghost



Logic

Provability theory

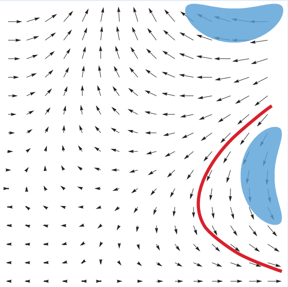
Math

Character-istic PDE

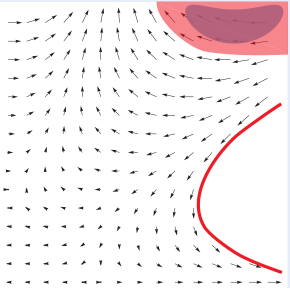
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

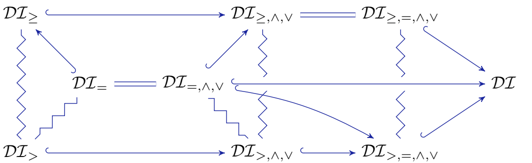
Differential Invariant



Differential Cut



Differential Ghost



Logic

Provability theory

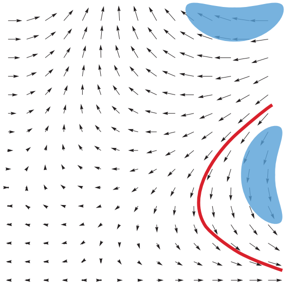
Math

Character-istic PDE

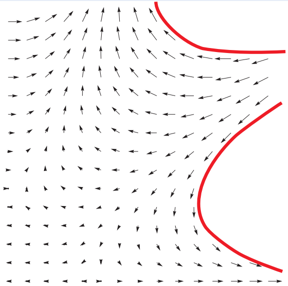
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

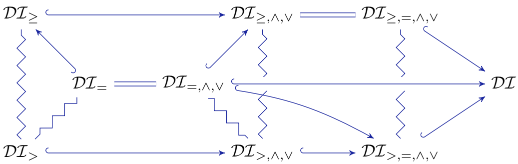
Differential Invariant



Differential Cut



Differential Ghost



Logic

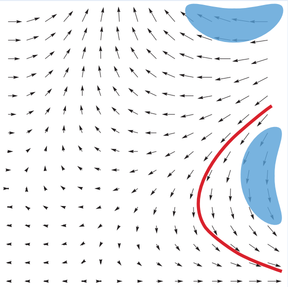
Provability theory

Math

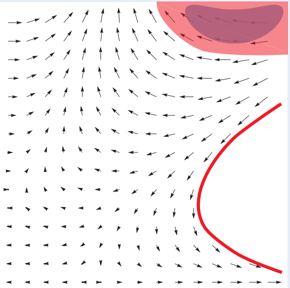
Character-istic PDE

JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

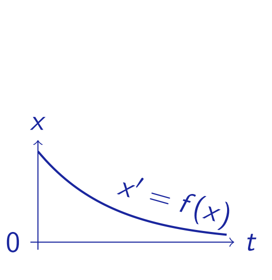
Differential Invariant

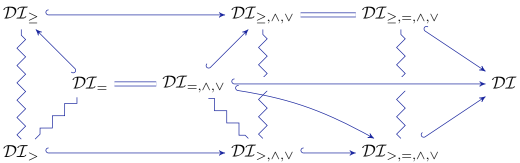


Differential Cut



Differential Ghost





Logic

Provability theory

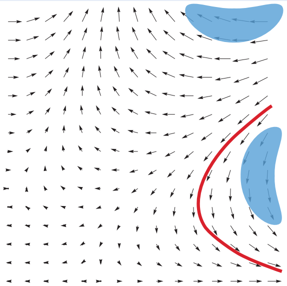
Math

Character-istic PDE

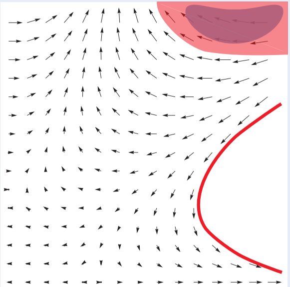
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

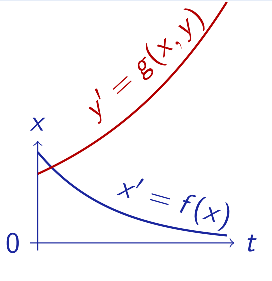
Differential Invariant

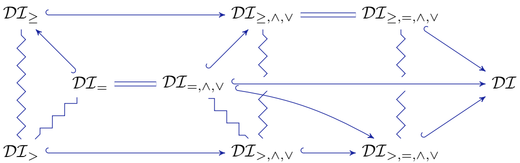


Differential Cut



Differential Ghost





Logic

Provability theory

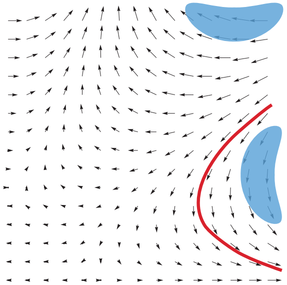
Math

Character-istic PDE

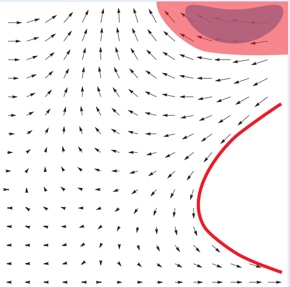
JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Invariants for Differential Equations

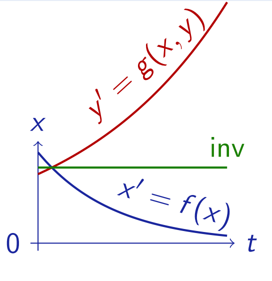
Differential Invariant

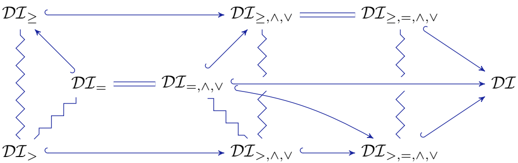


Differential Cut



Differential Ghost





Logic

Provability theory

Math

Characteristic PDE

JLogComput'10,CAV'08,FMSD'09,LMCS'12,LICS'12,ITP'12,CADE'15

Differential Equation Axioms & Differential Axioms

DW $[x' = f(x) \,\&\, Q]Q$

DC $\big([x' = f(x) \,\&\, Q]P \leftrightarrow [x' = f(x) \,\&\, Q \wedge R]P\big) \leftarrow [x' = f(x) \,\&\, Q]R$

DE $[x' = f(x) \,\&\, Q]P \leftrightarrow [x' = f(x) \,\&\, Q][x' := f(x)]P$

DI $\big([x' = f(x) \,\&\, Q]P \leftrightarrow [?Q]P\big) \leftarrow [x' = f(x) \,\&\, Q](P)'$

DG $[x' = f(x) \,\&\, Q]P \leftrightarrow \exists y\, [x' = f(x), y' = a(x)y + b(x) \,\&\, Q]P$

DS $[x' = c() \,\&\, Q]P \leftrightarrow \forall t{\geq}0\, \big((\forall 0{\leq}s{\leq}t\, q(x{+}c()s)) \rightarrow [x := x{+}c()t]P\big)$

$[':=]$ $[x' := e]p(x') \leftrightarrow p(e)$

$+'$ $(e + k)' = (e)' + (k)'$

$\cdot'$ $(e \cdot k)' = (e)' \cdot k + e \cdot (k)'$

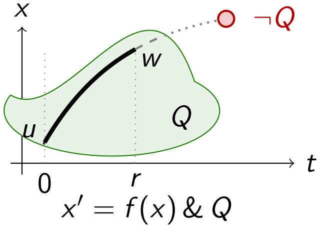
$\circ'$ $[y := g(x)][y' := 1]\big((f(g(x)))' = (f(y))' \cdot (g(x))'\big)$

CADE'15

Differential Equation Axioms

Axiom (Differential Weakening) (CADE'15)

DW $[x' = f(x) \,\&\, Q]Q$



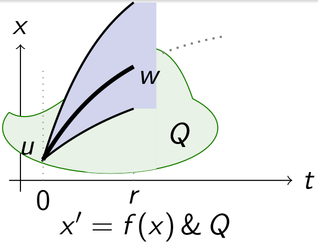
Differential equations cannot leave their evolution domains. Implies:

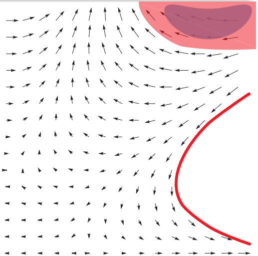
$$[x' = f(x) \,\&\, Q]P \leftrightarrow [x' = f(x) \,\&\, Q]\big(Q \rightarrow P\big)$$

Differential Equation Axioms

Axiom (Differential Cut) (CADE'15)

DC $\big([x' = f(x)\,\&\,Q]P \leftrightarrow [x' = f(x)\,\&\,Q \land R]P\big) \leftarrow [x' = f(x)\,\&\,Q]R$

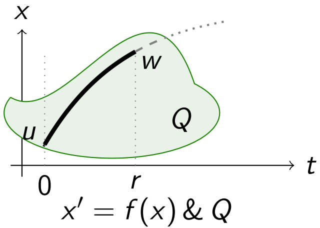


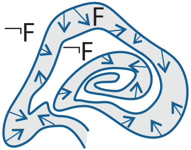


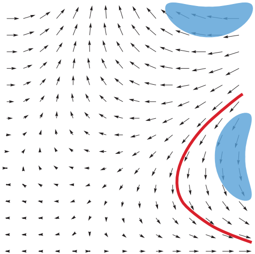
DC is a cut for differential equations.
DC is a differential modal modus ponens K.
Can't leave R, then might as well restrict state space to R.

Axiom (Differential Invariant) (CADE'15)

DI $([x' = f(x) \,\&\, Q]P \leftrightarrow [?Q]P) \leftarrow [x' = f(x) \,\&\, Q](P)'$





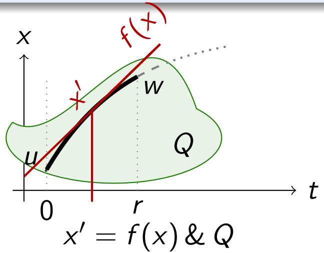


Differential invariant: if P true now and if differential $(P)'$ true always
What's the differential of a formula???
What's the meaning of a differential term . . . in a state???

Differential Equation Axioms

Axiom (Differential Effect) (CADE'15)

DE $[x' = f(x) \,\&\, Q]P \leftrightarrow [x' = f(x) \,\&\, Q][x' := f(x)]P$



Effect of differential equation on differential symbol x'

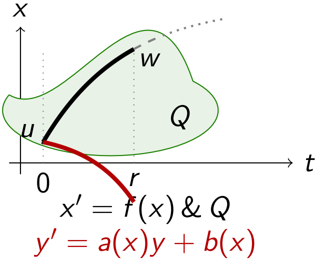
$[x' := f(x)]$ instantly mimics continuous effect $[x' = f(x)]$ on x'

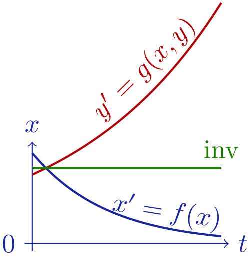
$[x' := f(x)]$ selects vector field $x' = f(x)$ for subsequent differentials

Differential Equation Axioms

Axiom (Differential Ghost) (CADE'15)

DG $[x' = f(x) \,\&\, Q]P \leftrightarrow \exists y\, [x' = f(x), y' = a(x)y + b(x) \,\&\, Q]P$





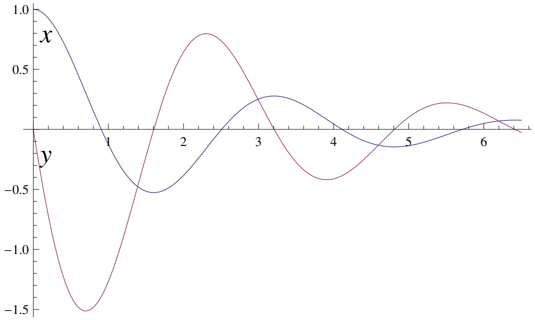
Differential ghost/auxiliaries: extra differential equations that exist

Can cause new invariants

"Dark matter" counterweight to balance conserved quantities

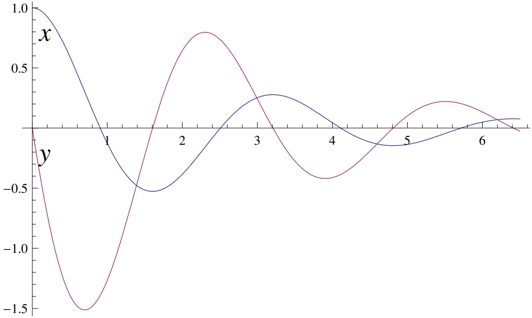
Differential Invariants for Differential Equations

$$\frac{}{\omega^2x^2+y^2\leq c^2 \vdash [x' = y, y' = -\omega^2 x - 2d\omega y \,\&\, (\omega\geq 0 \land d\geq 0)]\, \omega^2x^2+y^2\leq c^2}$$

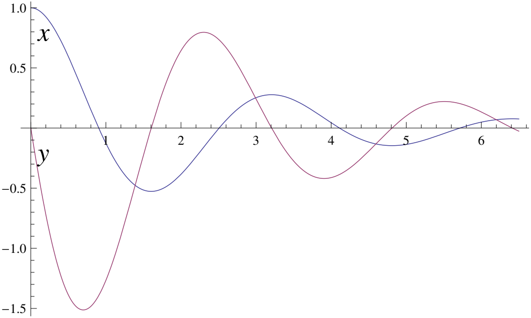


Differential Invariants for Differential Equations

$$\frac{\omega \geq 0 \land d \geq 0 \vdash [x' := y][y' := -\omega^2 x - 2d\omega y] 2\omega^2 x x' + 2 y y' \leq 0}{\omega^2 x^2 + y^2 \leq c^2 \vdash [x' = y, y' = -\omega^2 x - 2d\omega y \,\&\, (\omega \geq 0 \land d \geq 0)]\, \omega^2 x^2 + y^2 \leq c^2}$$

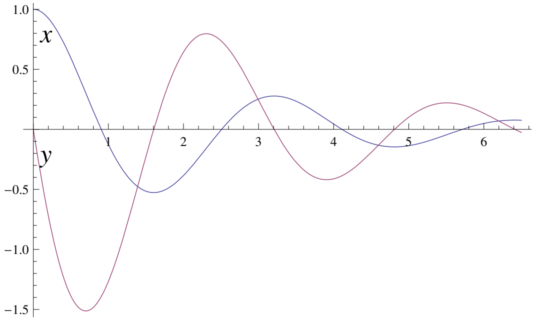


$$\dfrac{\dfrac{\omega\geq 0 \land d\geq 0 \vdash 2\omega^2 x y + 2y(-\omega^2 x - 2d\omega y) \leq 0}{\omega\geq 0 \land d\geq 0 \vdash [x':=y][y':=-\omega^2 x - 2d\omega y]2\omega^2 x x' + 2y y' \leq 0}}{\omega^2x^2+y^2\leq c^2 \vdash [x' = y, y' = -\omega^2 x - 2d\omega y \,\&\, (\omega\geq 0 \land d\geq 0)]\, \omega^2x^2+y^2\leq c^2}$$



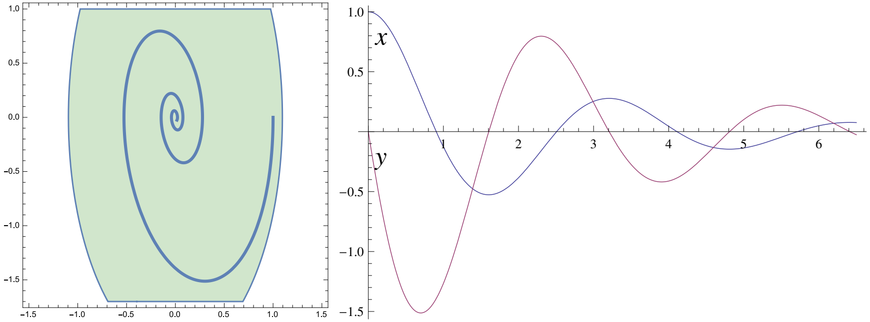
Differential Invariants for Differential Equations

$$\dfrac{\dfrac{\dfrac{*}{\omega\geq 0 \land d\geq 0 \vdash 2\omega^2 xy + 2y(-\omega^2 x - 2d\omega y) \leq 0}}{\omega\geq 0 \land d\geq 0 \vdash [x':=y][y':=-\omega^2 x - 2d\omega y] 2\omega^2 x x' + 2yy' \leq 0}}{\omega^2x^2+y^2\leq c^2 \vdash [x' = y, y' = -\omega^2 x - 2d\omega y \,\&\, (\omega\geq 0 \land d\geq 0)]\, \omega^2x^2+y^2\leq c^2}}$$



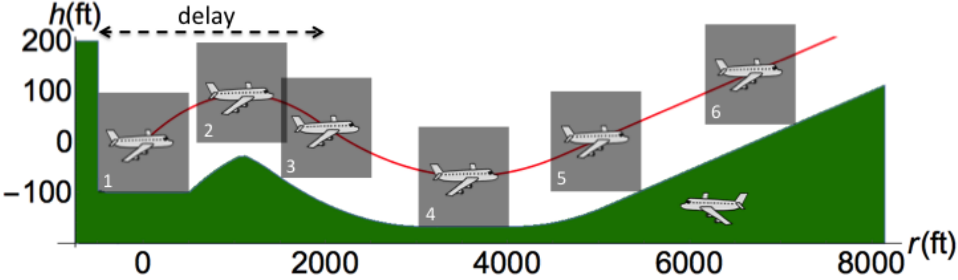
Differential Invariants for Differential Equations

$$\dfrac{\dfrac{\dfrac{*}{\omega\geq 0 \wedge d\geq 0 \vdash 2\omega^2 xy + 2y(-\omega^2 x - 2d\omega y) \leq 0}}{\omega\geq 0 \wedge d\geq 0 \vdash [x':=y][y':=-\omega^2 x - 2d\omega y]2\omega^2 x x' + 2yy' \leq 0}}{\omega^2x^2+y^2\leq c^2 \vdash [x' = y, y' = -\omega^2 x - 2d\omega y \,\&\, (\omega\geq 0 \wedge d\geq 0)]\, \omega^2x^2+y^2\leq c^2}$$



Airborne Collision Avoidance System ACAS X: Verify

- Developed by the FAA to replace current TCAS in aircraft
- Approximately optimizes Markov Decision Process on a grid
- Advisory from lookup tables with numerous 5D interpolation regions

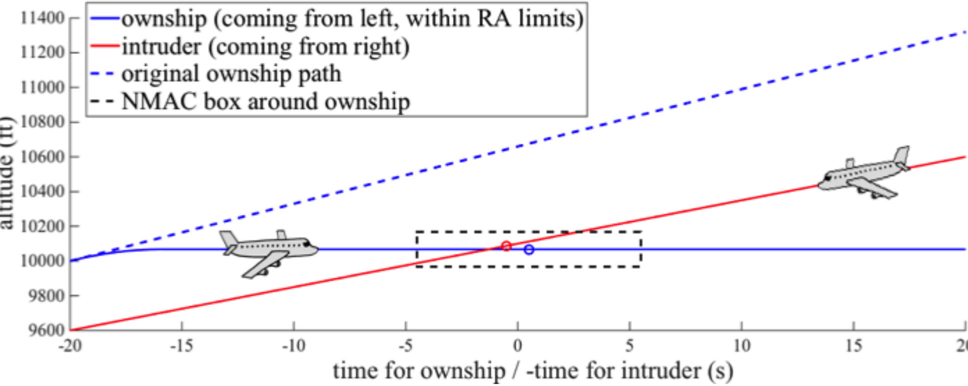


1. Identified safe region for each advisory symbolically
2. Proved safety for hybrid systems flight model in KeYmaera X

TACAS'15,EMSOFT'15

Airborne Collision Avoidance System ACAS X: Compare

ACAS X table comparison shows safe advisory in 97.7% of the 648,591,384,375 states compared (15,160,434,734 counterexamples).

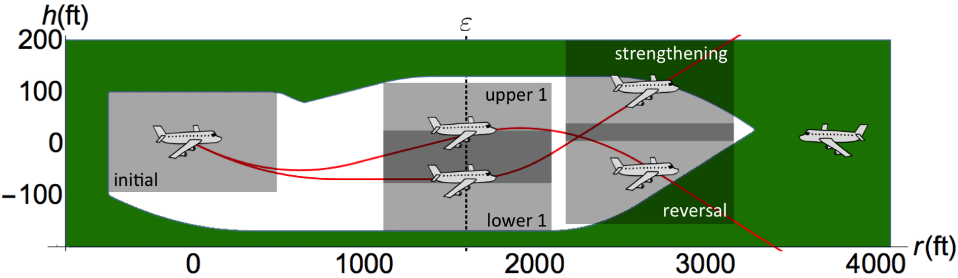


ACAS X issues DNC advisory, which induces collision unless corrected

TACAS'15,EMSOFT'15

Airborne Collision Avoidance System ACAS X: Refine

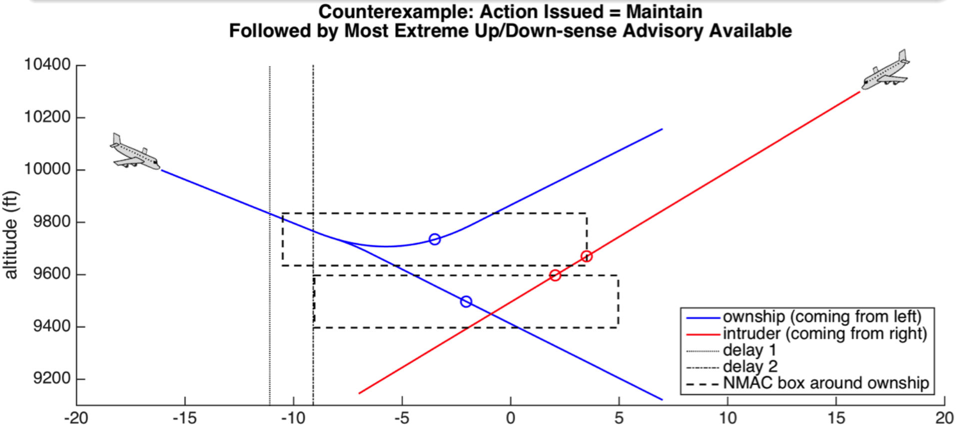
- Conservative, so too many counterexamples
- Settle for: safe for a little while with safe possible future
- Safeable advisory: a subsequent advisory can safely avoid collision



1. Identified safeable region for each advisory symbolically
2. Proved safety for hybrid systems flight model in KeYmaera X

STTT

ACAS X table comparison shows safeable advisory in more of the 648,591,384,375 states compared ($\approx$31.6 to 898.7 10^6 counterexamples).

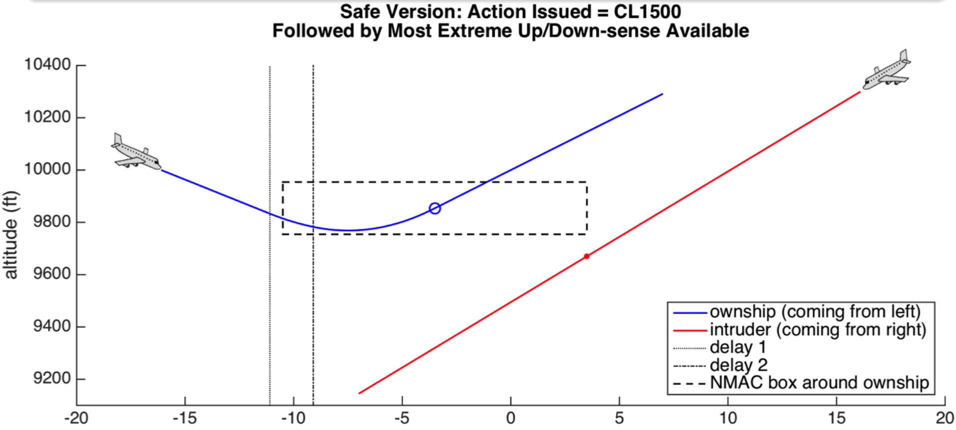


ACAS X issues Maintain advisory instead of CL1500

STTT

Airborne Collision Avoidance System ACAS X: Compare

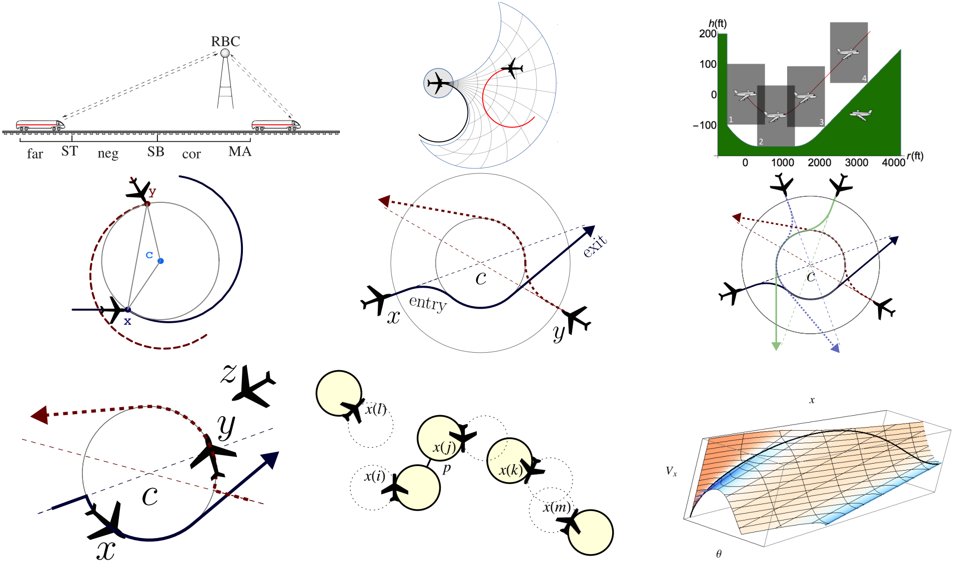
ACAS X table comparison shows safeable advisory in more of the 648,591,384,375 states compared ($\approx$31.6 to 898.7 10^6 counterexamples).



ACAS X issues Maintain advisory instead of CL1500

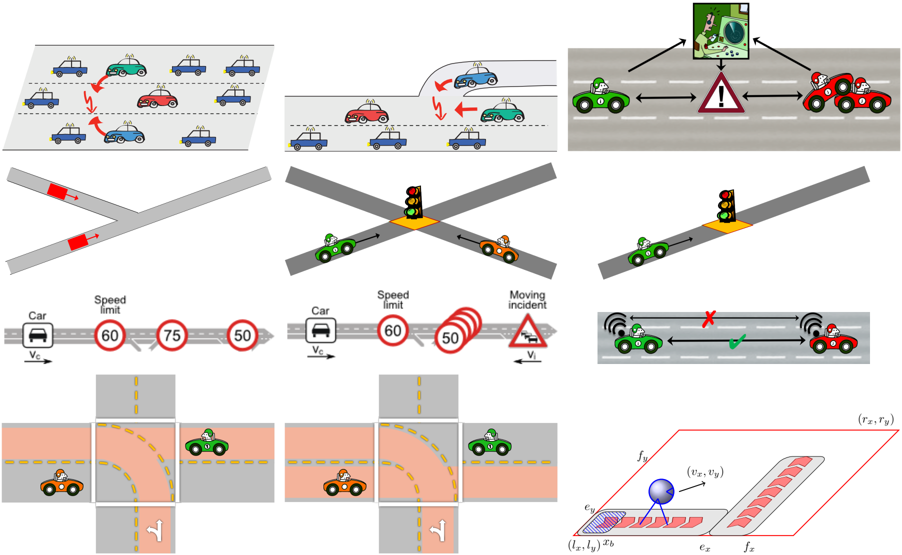
STTT

Verified CPS Applications



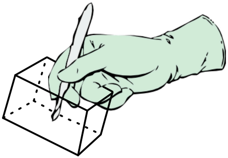
FEM'09,JAIS'14,TACAS'15,EMSOFT'15,CAV'08,FM'09,HSCC'11,HSCC'13,TACAS'14

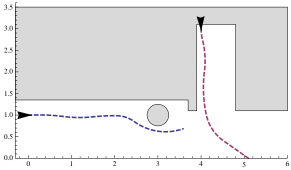
Verified CPS Applications

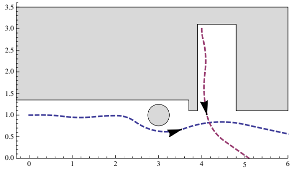


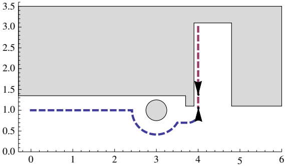
FM'11,LMCS'12,ICCPS'12,ITSC'11,ITSC'13,IJCAR'12

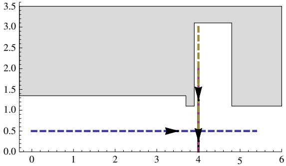
Verified CPS Applications

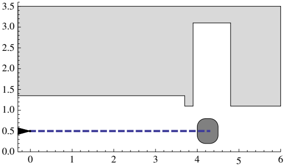


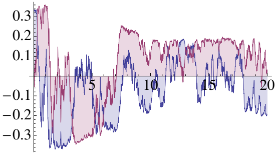


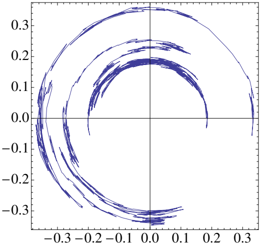


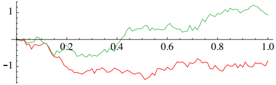




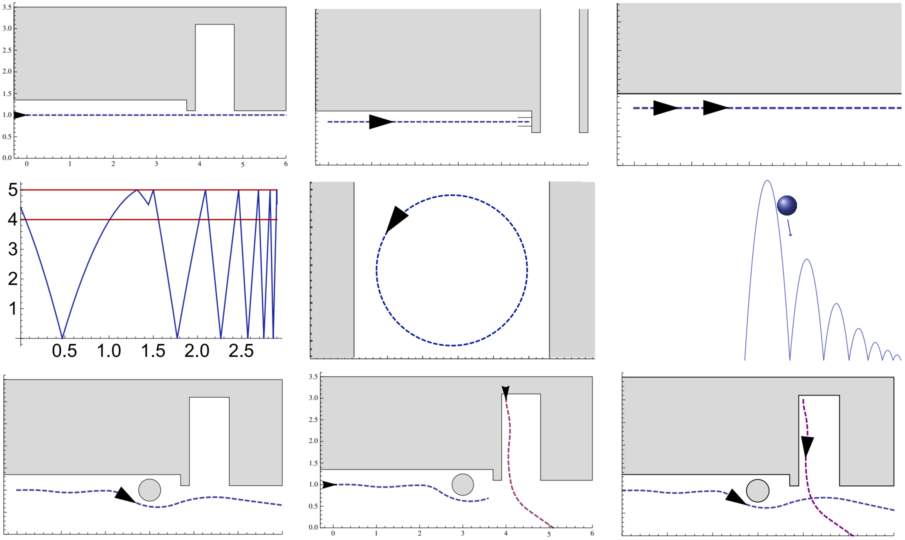






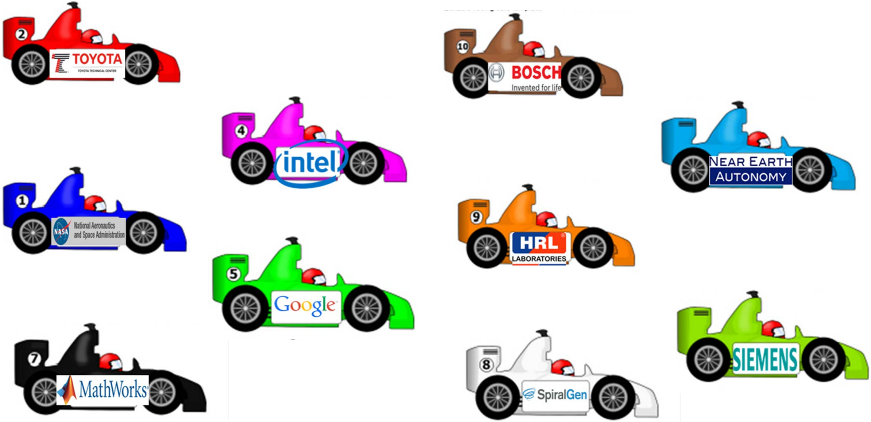


HSCC'13,RSS'13,CADE'12

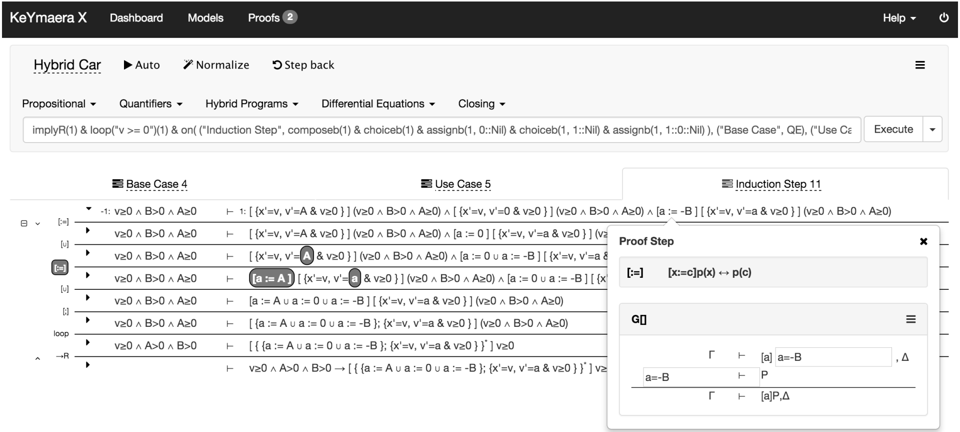


15-424/624/824 *Foundations of Cyber-Physical Systems* students

CPS V&V Grand Prix: Undergrad Course Competition



KeYmaera X http://keymaeraX.org/



CADE'15

An aXiomatic Tactical Theorem Prover for CPS

KeYmaera X http://keymaeraX.org/

Small Core	Increases trust, modularity, enables experimentation	(1652)
Tactics	Bridging between small core and powerful reasoning steps	(Hilbert) (Sequent)
Separation	Tactics can make courageous inferences Core establishes soundness	
Search&Do	Search-based tactics follow proof search strategies Constructive tactics directly build a proof	
Interaction	Interactive proofs mixed with tactical proofs and proof search	
Extensible	Flexible for new algorithms, new tactics, new logics, new proof rules, new axioms, ...	
Customize	Modular user interface, API	

CADE'15

KeYmaera X Microkernel for Soundness

	≈LOC		
KeYmaera X	1 652		hybrid prover
KeYmaera	65 989		hybrid prover
KeY	51 328		Java
Nuprl	15 000	+ 50 000	general math
MetaPRL	8 196		general math
Isabelle/Pure	8 113		general math
Coq	20 000		general math
HOL Light	396		general math
PHAVer	30 000		hybrid verifier
HSolver	20 000		hybrid verifier
SpaceEx	100 000		hybrid verifier
Flow*	25 000		hybrid verifier
dReal	50 000	+ millions	hybrid verifier
HyCreate2	6 081	+ user model analysis	hybrid verifier

Disclaimer: Self-reported estimates of the soundness-critical lines of code + rules

Uniform Substitution

Theorem (Soundness) replace all occurrences of $p(\cdot)$

$$(US)\ \frac{\phi}{\sigma(\phi)}$$

provided $FV(\sigma|_{\Sigma(\theta)}) \cap BV(\otimes(\cdot)) = \emptyset$ *for each operation* $\otimes(\theta)$ *in* ϕ

i.e. bound variables $U = \mathrm{BV}(\otimes(\cdot))$ of operator $\otimes$
are not free in the substitution on its argument θ (U-admissible)

$$\text{US}\ \frac{[a \cup b]p(\bar{x}) \leftrightarrow [a]p(\bar{x}) \land [b]p(\bar{x})}{[x := x+1 \cup x' = 1]x \geq 0 \leftrightarrow [x := x+1]x \geq 0 \land [x' = 1]x \geq 0}$$

Acknowledgments

Students and postdocs of the Logical Systems Lab at Carnegie Mellon
Brandon Bohrer, Nathan Fulton, David Henriques, Sarah Loos, João Martins
Erik Zawadzki, Khalil Ghorbal, Jean-Baptiste Jeannin, Stefan Mitsch

NSF DARPA DEPARTMENT OF TRANSPORTATION UNITED STATES OF AMERICA FEDERAL AVIATION ADMINISTRATION

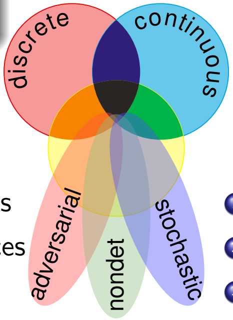
BOSCH Invented for life TOYOTA TOYOTA TECHNICAL CENTER APL JOHNS HOPKINS APPLIED PHYSICS LABORATORY

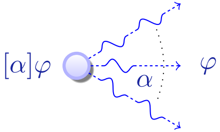
Logic & Proofs for Cyber-Physical Systems

Logical foundations make a big difference for CPS, and vice versa

differential dynamic logic

$$d\mathcal{L} = DL + HP$$





- Strong analytic foundations
- Practical reasoning advances
- Significant applications
- Catalyze many science areas

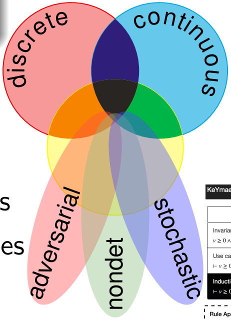
1. Multi-dynamical systems
2. Combine simple dynamics
3. Tame complexity
4. Complete axiomatization

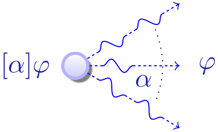
Numerous wonders remain to be discovered

Logical foundations make a big difference for CPS, and vice versa

differential dynamic logic

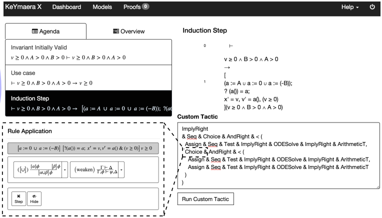
$$d\mathcal{L} = \text{DL} + \text{HP}$$





- Strong analytic foundations
- Practical reasoning advances
- Significant applications
- Catalyze many science areas

KeYmaera X

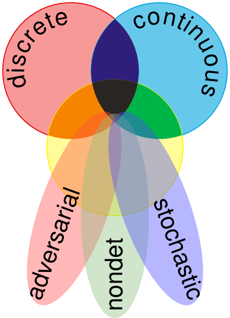


Numerous wonders remain to be discovered

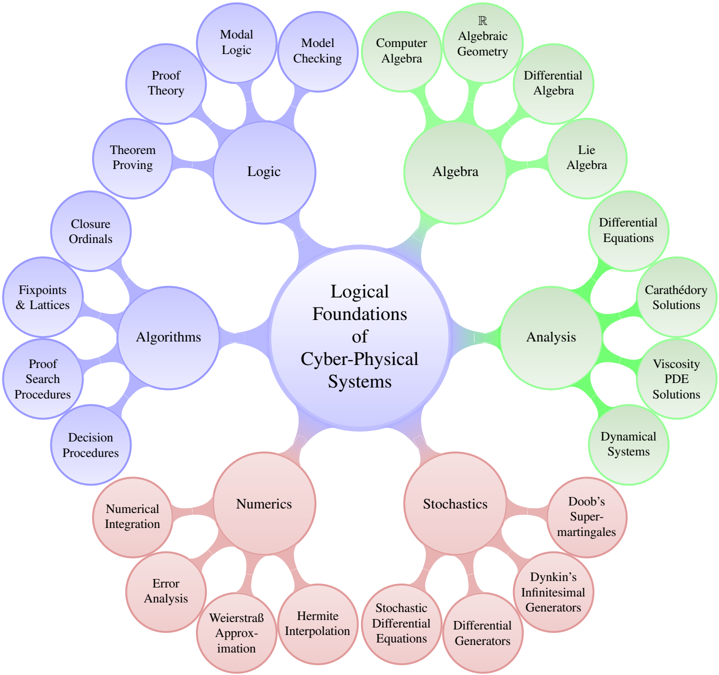
Future CPS Challenges

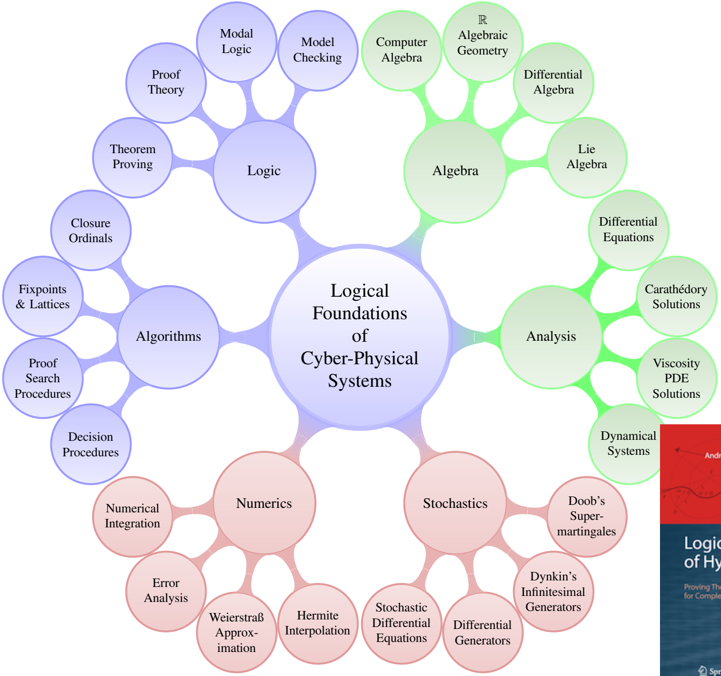
Numerous wonders remain to be discovered

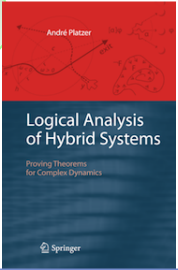
- Scalable continuous stochastics — CADE'11
- Concurrent CPS
- Real arithmetic: Scalable and verified — CADE'09
- Verified CPS implementations, ModelPlex — FMSD'16
- Correct CPS execution
- CPS-conducive tactic languages+libraries
- Tactics exploiting CPS structure/linearity/...
- Invariant generation
- Tactics & proofs for reachable set computations
- Parallel proof search & disprovers
- Correct model transformation
- Inspiring applications



CPSs deserve proofs as safety evidence!


Logical Foundations of Cyber-Physical Systems
Logic
Theorem Proving
Proof Theory
Modal Logic
Model Checking
Algebra
Computer Algebra
ℝ Algebraic Geometry
Differential Algebra
Lie Algebra
Analysis
Differential Equations
Carathédory Solutions
Viscosity PDE Solutions
Dynamical Systems
Stochastics
Doob's Super-martingales
Dynkin's Infinitesimal Generators
Differential Generators
Stochastic Differential Equations
Numerics
Hermite Interpolation
Weierstraß Approx-imation
Error Analysis
Numerical Integration
Algorithms
Decision Procedures
Proof Search Procedures
Fixpoints & Lattices
Closure Ordinals


Logical Foundations of Cyber-Physical Systems
Logic
Proof Theory
Modal Logic
Model Checking
Theorem Proving
Algebra
Computer Algebra
ℝ Algebraic Geometry
Differential Algebra
Lie Algebra
Analysis
Differential Equations
Carathédory Solutions
Viscosity PDE Solutions
Dynamical Systems
Algorithms
Closure Ordinals
Fixpoints & Lattices
Proof Search Procedures
Decision Procedures
Numerics
Numerical Integration
Error Analysis
Weierstraß Approx-imation
Hermite Interpolation
Stochastics
Doob's Super-martingales
Dynkin's Infinitesimal Generators
Differential Generators
Stochastic Differential Equations


André Platzer
Logical Analysis of Hybrid Systems
Proving Theorems for Complex Dynamics
Springer

Differential Dynamic Logic d$\mathcal{L}$: Semantics

Definition (Hybrid program semantics) ($[\![\cdot]\!] : \mathrm{HP} \to \wp(\mathcal{S} \times \mathcal{S})$)

$$
\begin{aligned}
[\![x := e]\!] &= \{(\omega, \nu) \;:\; \nu = \omega \text{ except } [\![x]\!]\nu = [\![e]\!]\omega\} \\
[\![?Q]\!] &= \{(\omega, \omega) \;:\; \omega \in [\![Q]\!]\} \\
[\![x' = f(x)]\!] &= \{(\varphi(0), \varphi(r)) \;:\; \varphi \models x' = f(x) \text{ for some duration } r\} \\
[\![\alpha \cup \beta]\!] &= [\![\alpha]\!] \cup [\![\beta]\!] \\
[\![\alpha; \beta]\!] &= [\![\alpha]\!] \circ [\![\beta]\!] \\
[\![\alpha^*]\!] &= \bigcup_{n \in \mathbb{N}} [\![\alpha^n]\!]
\end{aligned}
$$

Definition (d$\mathcal{L}$ semantics) ($[\![\cdot]\!] : \mathrm{Fml} \to \wp(\mathcal{S})$)

$$
\begin{aligned}
[\![e \geq \tilde{e}]\!] &= \{\omega \;:\; [\![e]\!]\omega \geq [\![\tilde{e}]\!]\omega\} \\
[\![\neg P]\!] &= [\![P]\!]^{\complement} \\
[\![P \wedge Q]\!] &= [\![P]\!] \cap [\![Q]\!] \\
[\![\langle\alpha\rangle P]\!] &= [\![\alpha]\!] \circ [\![P]\!] = \{\omega \;:\; \nu \in [\![P]\!] \text{ for some } \nu :\; (\omega, \nu) \in [\![\alpha]\!]\} \\
[\![[\alpha]P]\!] &= [\![\neg\langle\alpha\rangle\neg P]\!] = \{\omega \;:\; \nu \in [\![P]\!] \text{ for all } \nu :\; (\omega, \nu) \in [\![\alpha]\!]\} \\
[\![\exists x\, P]\!] &= \{\omega \;:\; \omega_x^r \in [\![P]\!] \text{ for some } r \in \mathbb{R}\}
\end{aligned}
$$

André Platzer.
Logic & proofs for cyber-physical systems.
In Nicola Olivetti and Ashish Tiwari, editors, *IJCAR*, volume 9706 of *LNCS*, pages 15–21. Springer, 2016.
doi:10.1007/978-3-319-40229-1_3.

André Platzer.
Logics of dynamical systems.
In LICS [27], pages 13–24.
doi:10.1109/LICS.2012.13.

André Platzer.
Differential dynamic logic for hybrid systems.
J. Autom. Reas., 41(2):143–189, 2008.
doi:10.1007/s10817-008-9103-8.

André Platzer.
A uniform substitution calculus for differential dynamic logic.
In Amy Felty and Aart Middeldorp, editors, *CADE*, volume 9195 of *LNCS*, pages 467–481. Springer, 2015.

doi:10.1007/978-3-319-21401-6_32.

André Platzer.
Differential game logic.
ACM Trans. Comput. Log., 17(1):1:1–1:51, 2015.
doi:10.1145/2817824.

André Platzer.
The complete proof theory of hybrid systems.
In LICS [27], pages 541–550.
doi:10.1109/LICS.2012.64.

André Platzer.
A complete axiomatization of quantified differential dynamic logic for distributed hybrid systems.
Log. Meth. Comput. Sci., 8(4):1–44, 2012.
Special issue for selected papers from CSL'10.
doi:10.2168/LMCS-8(4:17)2012.

André Platzer.
Stochastic differential dynamic logic for stochastic hybrid programs.

In Nikolaj Bjørner and Viorica Sofronie-Stokkermans, editors, *CADE*, volume 6803 of *LNCS*, pages 431–445. Springer, 2011.
doi:10.1007/978-3-642-22438-6_34.

André Platzer.
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 20(1):309–352, 2010.
doi:10.1093/logcom/exn070.

André Platzer and Edmund M. Clarke.
Computing differential invariants of hybrid systems as fixedpoints.
In Aarti Gupta and Sharad Malik, editors, *CAV*, volume 5123 of *LNCS*, pages 176–189. Springer, 2008.
doi:10.1007/978-3-540-70545-1_17.

André Platzer and Edmund M. Clarke.
Computing differential invariants of hybrid systems as fixedpoints.
Form. Methods Syst. Des., 35(1):98–120, 2009.
Special issue for selected papers from CAV'08.
doi:10.1007/s10703-009-0079-8.

André Platzer.
The structure of differential invariants and differential cut elimination.
Log. Meth. Comput. Sci., 8(4):1–38, 2012.
doi:10.2168/LMCS-8(4:16)2012.

André Platzer.
A differential operator approach to equational differential invariants.
In Lennart Beringer and Amy Felty, editors, *ITP*, volume 7406 of *LNCS*, pages 28–48. Springer, 2012.
doi:10.1007/978-3-642-32347-8_3.

Jean-Baptiste Jeannin, Khalil Ghorbal, Yanni Kouskoulas, Ryan Gardner, Aurora Schmidt, Erik Zawadzki, and André Platzer.
A formally verified hybrid system for the next-generation airborne collision avoidance system.
In Christel Baier and Cesare Tinelli, editors, *TACAS*, volume 9035 of *LNCS*, pages 21–36. Springer, 2015.
doi:10.1007/978-3-662-46681-0_2.

Jean-Baptiste Jeannin, Khalil Ghorbal, Yanni Kouskoulas, Ryan Gardner, Aurora Schmidt, Erik Zawadzki, and André Platzer.
Formal verification of ACAS X, an industrial airborne collision avoidance system.
In Alain Girault and Nan Guan, editors, *EMSOFT*, pages 127–136. IEEE, 2015.
doi:10.1109/EMSOFT.2015.7318268.

Jean-Baptiste Jeannin, Khalil Ghorbal, Yanni Kouskoulas, Aurora Schmidt, Ryan Gardner, Stefan Mitsch, and André Platzer.
A formally verified hybrid system for safe advisories in the next-generation airborne collision avoidance system, 2015.
http://www.cs.cmu.edu/~aplatzer/pub/acasx-long.pdf.

André Platzer.
Foundations of cyber-physical systems.
Lecture Notes 15-424/624, Carnegie Mellon University, 2016.
URL: http://www.cs.cmu.edu/~aplatzer/course/fcps16.html.

Stefan Mitsch and André Platzer.

ModelPlex: Verified runtime validation of verified cyber-physical system models.
Form. Methods Syst. Des., 2016.
Special issue of selected papers from RV'14.
doi:10.1007/s10703-016-0241-z.

André Platzer.
Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics.
Springer, Heidelberg, 2010.
doi:10.1007/978-3-642-14509-4.

Thomas A. Henzinger.
The theory of hybrid automata.
In *LICS*, pages 278–292, Los Alamitos, 1996. IEEE Computer Society.
doi:10.1109/LICS.1996.561342.

Jennifer M. Davoren and Anil Nerode.
Logics for hybrid systems.
IEEE, 88(7):985–1010, July 2000.

Ashish Tiwari.
Abstractions for hybrid systems.
Form. Methods Syst. Des., 32(1):57–83, 2008.
doi:10.1007/s10703-007-0044-3.

Jan Lunze and Françoise Lamnabhi-Lagarrigue, editors.
Handbook of Hybrid Systems Control: Theory, Tools, Applications.
Cambridge Univ. Press, 2009.

Paulo Tabuada.
Verification and Control of Hybrid Systems: A Symbolic Approach.
Springer, 2009.

Rajeev Alur.
Principles of Cyber-Physical Systems.
MIT Press, 2015.

Laurent Doyen, Goran Frehse, George J. Pappas, and André Platzer.
Verification of hybrid systems.
In Edmund M. Clarke, Thomas A. Henzinger, and Helmut Veith, editors, *Handbook of Model Checking*, chapter 28. Springer, 2017.

Proceedings of the 27th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25–28, 2012. IEEE, 2012.