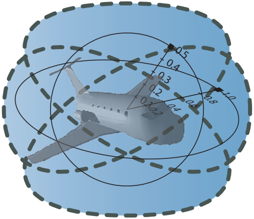
Differential Game Logic

André Platzer

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



Outline

1. CPS Applications
2. Differential Game Logic
   - Differential Hybrid Games
   - Denotational Semantics
   - Determinacy
3. Proofs for CPS
   - Axiomatization
   - Soundness and Completeness
   - Corollaries
   - Separating Axioms
4. Expressiveness
5. Summary

Can you trust a computer to control physics?

Can you trust a computer to control physics?

Rationale

1. Safety guarantees require analytic foundations.
2. Foundations revolutionized digital computer science & our society.
3. Need even stronger foundations when software reaches out into our physical world.

How can we provide people with cyber-physical systems they can bet their lives on? — Jeannette Wing

Cyber-physical Systems

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

Outline

1. CPS Applications

2. Differential Game Logic
   - Differential Hybrid Games
   - Denotational Semantics
   - Determinacy

3. Proofs for CPS
   - Axiomatization
   - Soundness and Completeness
   - Corollaries
   - Separating Axioms

4. Expressiveness

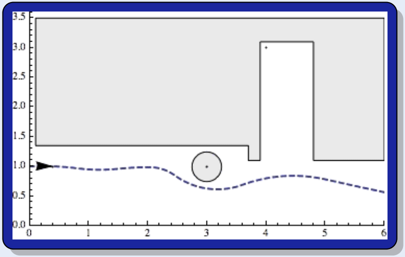
5. Summary

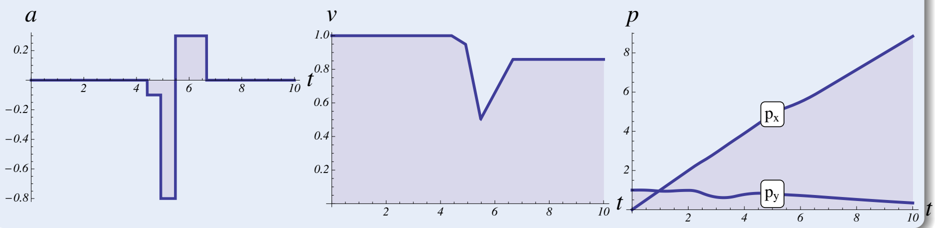
CPS Analysis: Robot Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)



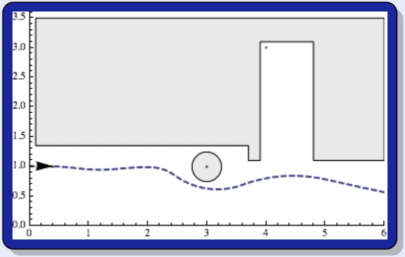


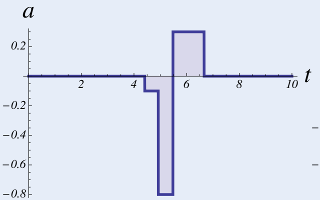
CPS Analysis: Robot Control

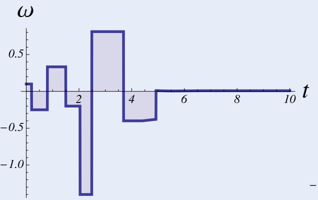
Challenge (Hybrid Systems)

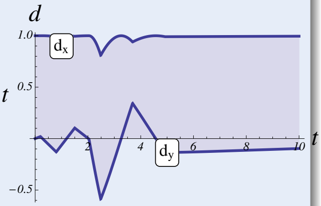
Fixed rule describing state evolution with both

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)







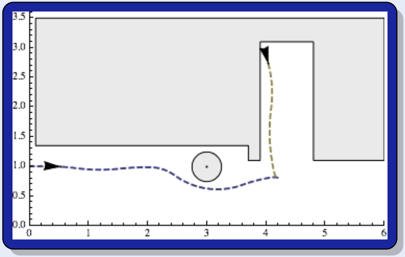


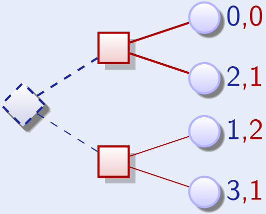
CPS Analysis: Robot Control

Challenge (Games)

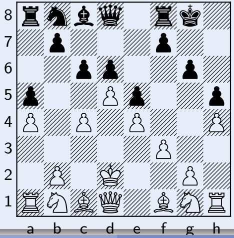
Game rules describing play evolution with both

- Angelic choices (player ◇ Angel)
- Demonic choices (player □ Demon)





◇\□	Tr	Pl
Trash	1,2	0,0
Plant	0,0	2,1

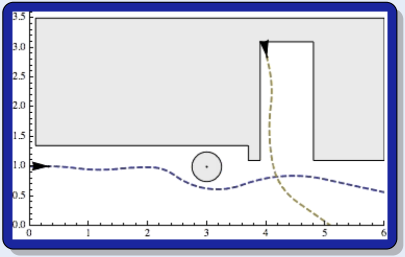


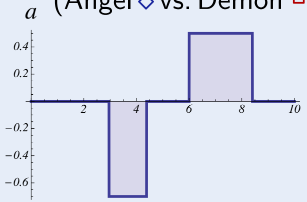
CPS Analysis: Robot Control

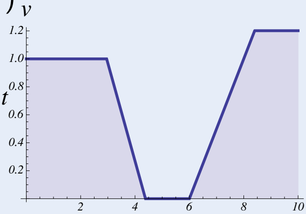
Challenge (Hybrid Games)

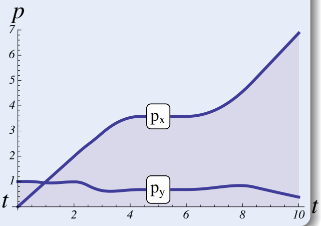
Game rules describing play evolution with

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)
- Adversarial dynamics (Angel ◇ vs. Demon □)







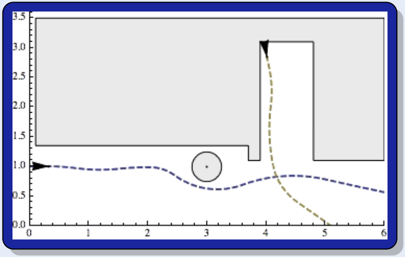


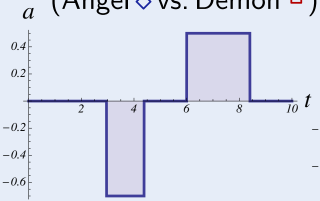
CPS Analysis: Robot Control

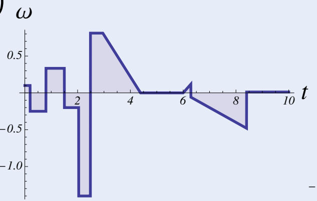
Challenge (Hybrid Games)

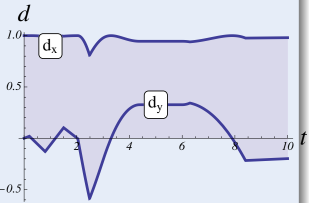
Game rules describing play evolution with

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)
- Adversarial dynamics (Angel ◇ vs. Demon □)







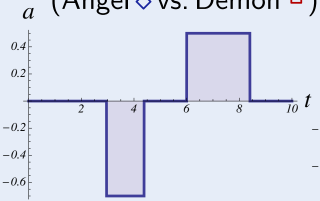


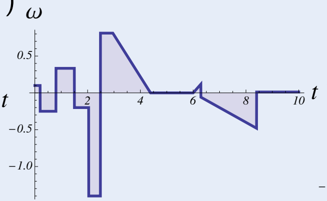
CPS Analysis: RoboCup Soccer

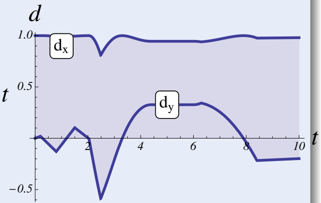
Challenge (Hybrid Games)

Game rules describing play evolution with

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)
- Adversarial dynamics (Angel ◇ vs. Demon □)







Contributions

Logical foundations for hybrid games

1. Compositional programming language for hybrid games
2. Compositional logic and proof calculus for winning strategy existence
3. Hybrid games determined
4. Winning region computations terminate after $\geq\omega_1^{\mathrm{CK}}$ iterations
5. Separate truth ($\exists$ winning strategy) vs. proof (winning certificate) vs. proof search (automatic construction)
6. Sound & relatively complete
7. Expressiveness
8. Fragments quite successful in applications
9. Generalizations in logic enable more applications

Outline

1. CPS Applications
2. Differential Game Logic
   - Differential Hybrid Games
   - Denotational Semantics
   - Determinacy
3. Proofs for CPS
   - Axiomatization
   - Soundness and Completeness
   - Corollaries
   - Separating Axioms
4. Expressiveness
5. Summary

Differential Game Logic dG$\mathcal{L}$: Syntax

Definition (Hybrid game a)

$$x := f(x) \mid ?Q \mid x' = f(x) \mid a \cup b \mid a; b \mid a^* \mid a^d$$

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

$$p(e_1, \dots, e_n) \mid e_1 \geq e_2 \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle a \rangle P \mid [a]P$$

TOCL'15

Differential Game Logic d$G\mathcal{L}$: Syntax

Discrete Assign

Test Game

Differential Equation

Choice Game

Seq. Game

Repeat Game

Definition (Hybrid game a)

$$x := f(x) \mid ?Q \mid x' = f(x) \mid a \cup b \mid a; b \mid a^* \mid a^d$$

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

$$p(e_1, \ldots, e_n) \mid e_1 \geq e_2 \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle a \rangle P \mid [a]P$$

All Reals

Some Reals

TOCL'15

Differential Game Logic dG$\mathcal{L}$: Syntax

Discrete Assign

Test Game

Differential Equation

Choice Game

Seq. Game

Repeat Game

Dual Game

Definition (Hybrid game a)

$$x := f(x) \mid ?Q \mid x' = f(x) \mid a \cup b \mid a; b \mid a^* \mid a^d$$

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

$$p(e_1, \ldots, e_n) \mid e_1 \geq e_2 \mid \neg P \mid P \land Q \mid \forall x\, P \mid \exists x\, P \mid \langle a \rangle P \mid [a]P$$

All Reals

Some Reals

TOCL'15

Differential Game Logic d$G\mathcal{L}$: Syntax

Discrete Assign

Test Game

Differential Equation

Choice Game

Seq. Game

Repeat Game

Dual Game

Definition (Hybrid game a)

$$x := f(x) \mid ?Q \mid x' = f(x) \mid a \cup b \mid a; b \mid a^* \mid a^d$$

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

$$p(e_1, \dots, e_n) \mid e_1 \geq e_2 \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle a \rangle P \mid [a]P$$

All Reals

Some Reals

Angel Wins

TOCL'15

Differential Game Logic d$G\mathcal{L}$: Syntax

Discrete Assign | Test Game | Differential Equation | Choice Game | Seq. Game | Repeat Game | Dual Game

Definition (Hybrid game a)

$$x := f(x) \mid ?Q \mid x' = f(x) \mid a \cup b \mid a; b \mid a^* \mid a^d$$

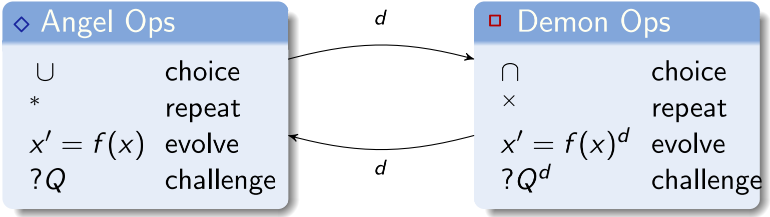
Definition (d$G\mathcal{L}$ Formula P)

$$p(e_1, \ldots, e_n) \mid e_1 \geq e_2 \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle a \rangle P \mid [a]P$$

All Reals | Some Reals | Angel Wins | Demon Wins

TOCL'15

Definable Game Operators



◇ Angel Ops

$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

□ Demon Ops

$\cap$	choice
$\times$	repeat
$x' = f(x)^d$	evolve
$?Q^d$	challenge

$$\texttt{if}(Q)\, a\, \texttt{else}\, b \equiv (?Q; a) \cup (?\neg Q; b)$$

$$\texttt{while}(Q)\, a \equiv (?Q; a)^*; ?\neg Q$$

$$a \cap b \equiv (a^d \cup b^d)^d$$

$$a^\times \equiv ((a^d)^*)^d$$

$$(x' = f(x) \,\&\, Q)^d \not\equiv x' = f(x) \,\&\, Q$$

$$(x := f(x))^d \equiv x := f(x)$$

$$?Q^d \not\equiv ?Q$$

Simple Examples

$$\langle (x := x+1; (x' = x^2)^d \cup x := x-1)^* \rangle\, (0 \le x < 1)$$

$$\langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \le x < 1)$$

$$\begin{aligned}
&(w-e)^2 \le 1 \land v = f \rightarrow \\
&\Big\langle \big((u := 1 \cap u := -1); \\
&\quad (g := 1 \cup g := -1); \\
&\quad t := 0; \\
&\quad (w' = v, v' = u, e' = f, f' = g, t' = 1 \,\&\, t \le 1)^d \\
&\big)^\times \Big\rangle\, (w-e)^2 \le 1
\end{aligned}$$

Simple Examples

$$\vDash \langle (x := x+1; (x' = x^2)^d \cup x := x-1)^* \rangle\, (0 \leq x < 1)$$

$$\langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \leq x < 1)$$

$$\begin{aligned}
&(w-e)^2 \leq 1 \land v = f \rightarrow \\
&\big\langle \big((u := 1 \cap u := -1); \\
&\quad (g := 1 \cup g := -1); \\
&\quad t := 0; \\
&\quad (w' = v, v' = u, e' = f, f' = g, t' = 1 \,\&\, t \leq 1)^d \\
&\big)^\times \big\rangle\, (w-e)^2 \leq 1
\end{aligned}$$

Simple Examples

$$\vDash \langle (x := x+1; (x' = x^2)^d \cup x := x-1)^* \rangle \, (0 \le x < 1)$$

$$\nvDash \langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \le x < 1)$$

$$\begin{aligned}
&(w-e)^2 \le 1 \land v = f \to \\
&\Big\langle \big((u := 1 \cap u := -1); \\
&\quad (g := 1 \cup g := -1); \\
&\quad t := 0; \\
&\quad (w' = v, v' = u, e' = f, f' = g, t' = 1 \,\&\, t \le 1)^d \\
&\big)^\times \Big\rangle \, (w-e)^2 \le 1
\end{aligned}$$

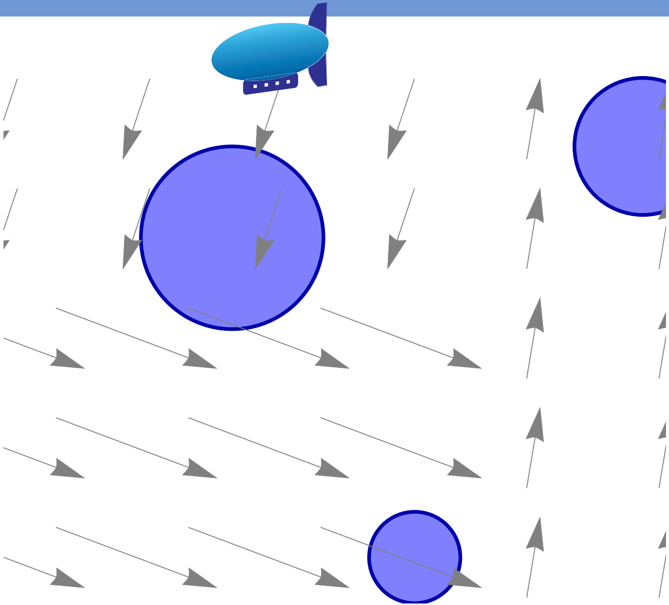
Simple Examples

$$\vDash \langle (x := x+1; (x' = x^2)^d \cup x := x-1)^* \rangle\, (0 \le x < 1)$$

$$\nvDash \langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \le x < 1)$$

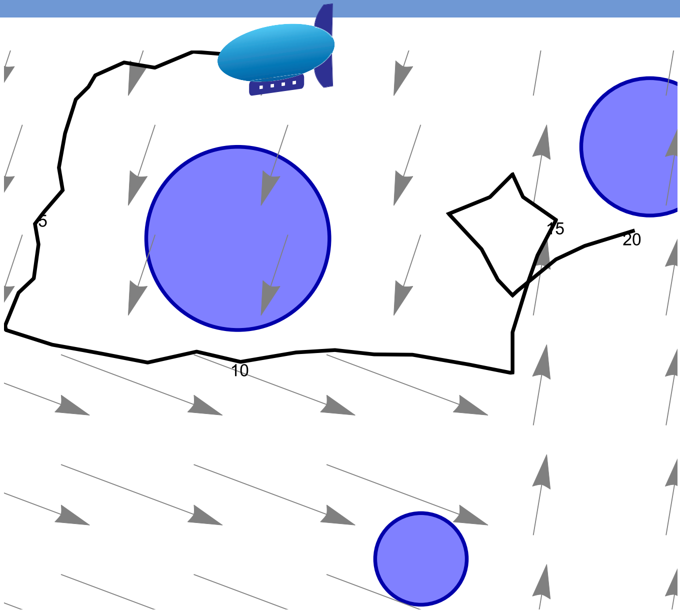
$$\begin{aligned}
\vDash\ & (w-e)^2 \le 1 \land v = f \to \\
& \Big\langle \Big((u := 1 \cap u := -1); \\
& \quad (g := 1 \cup g := -1); \\
& \quad t := 0; \\
& \quad (w' = v, v' = u, e' = f, f' = g, t' = 1 \,\&\, t \le 1)^d \\
& \Big)^\times \Big\rangle\, (w-e)^2 \le 1
\end{aligned}$$

Differential Hybrid Games: Zeppelin Obstacle Parcours



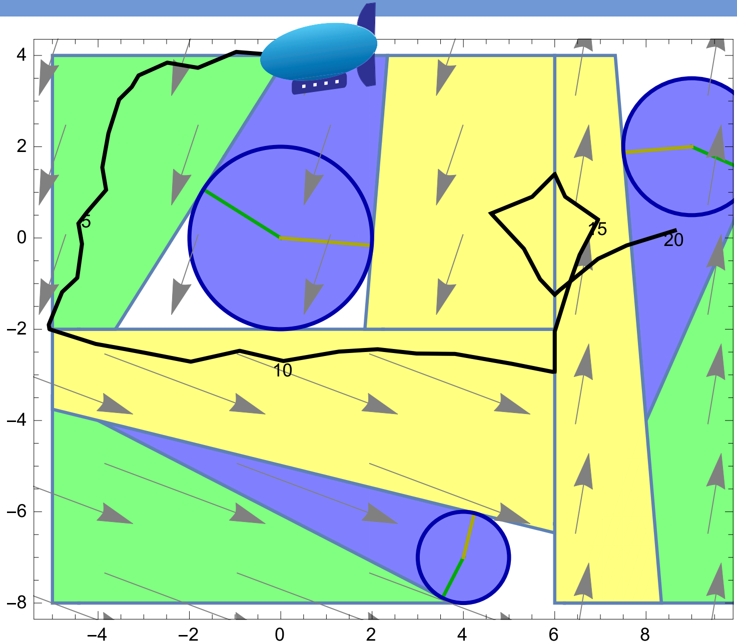
arXiv

Differential Hybrid Games: Zeppelin Obstacle Parcours



arXiv

Differential Hybrid Games: Zeppelin Obstacle Parcours



arXiv

Differential Game Invariants

Theorem (Differential Game Invariants)

$$\text{(DGI)} \quad \frac{\exists y \in Y\, \forall z \in Z\, F'^{\,f(x,y,z)}_{x'}}{F \rightarrow [x' = f(x,y,z) \&^{d} y \in Y \& z \in Z]F}$$

arXiv

Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

$$
\begin{aligned}
\varsigma_{x:=f(x)}(X) &= \{s \in \mathcal{S} \;:\; s_x^{\llbracket f(x) \rrbracket_s} \in X\}\\
\varsigma_{x'=f(x)}(X) &= \{\varphi(0) \in \mathcal{S} \;:\; \varphi(r) \in X, \frac{\mathrm{d}\,\varphi(t)(x)}{\mathrm{d}t}(\zeta) = \llbracket f(x) \rrbracket_{\varphi(\zeta)} \text{ for all } \zeta\}\\
\varsigma_{?P}(X) &= \llbracket P \rrbracket \cap X\\
\varsigma_{a \cup b}(X) &= \varsigma_a(X) \cup \varsigma_b(X)\\
\varsigma_{a;b}(X) &= \varsigma_a(\varsigma_b(X))\\
\varsigma_{a^*}(X) &= \bigcap\{Z \subseteq \mathcal{S} \;:\; X \cup \varsigma_a(Z) \subseteq Z\}\\
\varsigma_{a^d}(X) &= (\varsigma_a(X^\complement))^\complement
\end{aligned}
$$

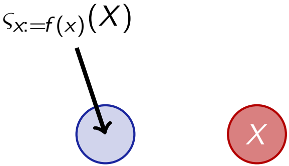
Definition (dG$\mathcal{L}$ Formula P)

$$
\begin{aligned}
\llbracket e_1 \geq e_2 \rrbracket &= \{s \in \mathcal{S} \;:\; \llbracket e_1 \rrbracket_s \geq \llbracket e_2 \rrbracket_s\}\\
\llbracket \neg P \rrbracket &= (\llbracket P \rrbracket)^\complement\\
\llbracket P \land Q \rrbracket &= \llbracket P \rrbracket \cap \llbracket Q \rrbracket\\
\llbracket \langle a \rangle P \rrbracket &= \varsigma_a(\llbracket P \rrbracket)\\
\llbracket [a] P \rrbracket &= \delta_a(\llbracket P \rrbracket)
\end{aligned}
$$

Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

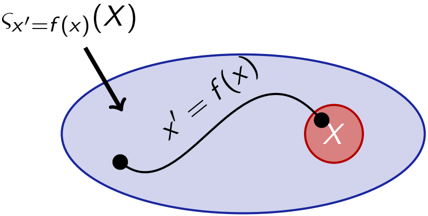
$$\varsigma_{x:=f(x)}(X) = \{s \in \mathcal{S} : s_x^{[\![f(x)]\!]_s} \in X\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

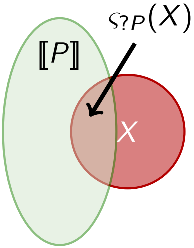
$$\varsigma_{x'=f(x)}(X) = \{\varphi(0) \in \mathcal{S} : \varphi(r) \in X, \frac{\mathrm{d}\,\varphi(t)(x)}{\mathrm{d}t}(\zeta) = [\![f(x)]\!]_{\varphi(\zeta)} \text{ for all } \zeta\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

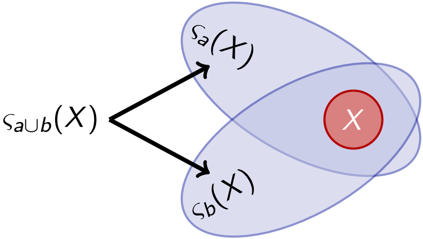
$$\varsigma_{?P}(X) = [\![P]\!] \cap X$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

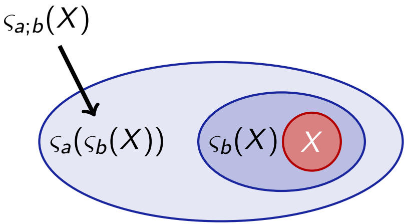
$$\varsigma_{a\cup b}(X) = \varsigma_a(X) \cup \varsigma_b(X)$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

$$\varsigma_{a;b}(X) = \varsigma_a(\varsigma_b(X))$$



Differential Game Logic: Denotational Semantics

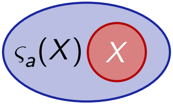
Definition (Hybrid game *a*: denotational semantics)

$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} \ : \ X \cup \varsigma_a(Z) \subseteq Z\}$$

Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

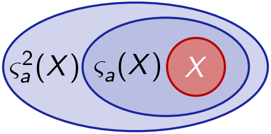
$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} \ : \ X \cup \varsigma_a(Z) \subseteq Z\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

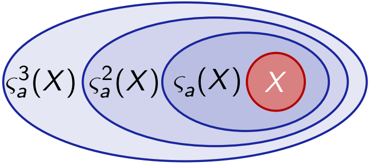
$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} \; : \; X \cup \varsigma_a(Z) \subseteq Z\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

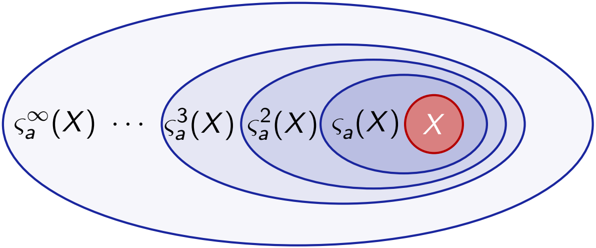
$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} : X \cup \varsigma_a(Z) \subseteq Z\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

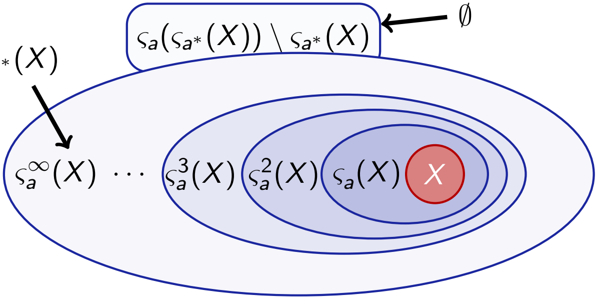
$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} \;:\; X \cup \varsigma_a(Z) \subseteq Z\}$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

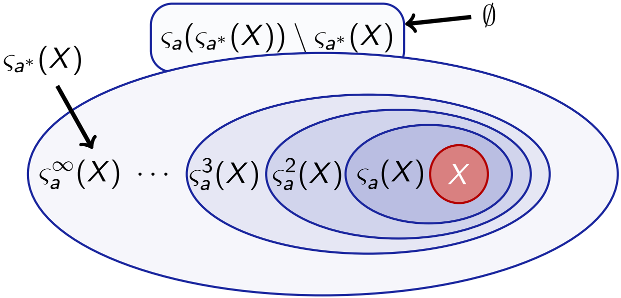
$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} : X \cup \varsigma_a(Z) \subseteq Z\}$$

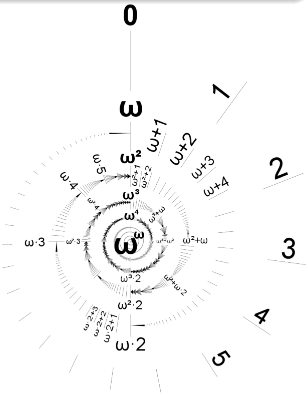

$\varsigma_a(\varsigma_{a^*}(X)) \setminus \varsigma_{a^*}(X)$
$\emptyset$
$\varsigma_{a^*}(X)$
$\varsigma_a^\infty(X)$ ⋯
$\varsigma_a^3(X)$
$\varsigma_a^2(X)$
$\varsigma_a(X)$
X

Differential Game Logic: Denotational Semantics

Definition (Hybrid game a: denotational semantics)

$$\varsigma_{a^*}(X) = \bigcap\{Z \subseteq \mathcal{S} : X \cup \varsigma_a(Z) \subseteq Z\}$$



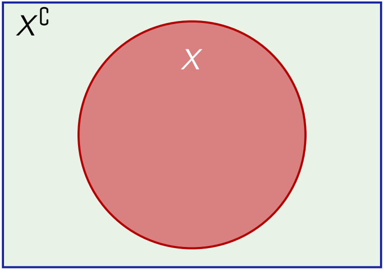


$\geq \omega_1^{CK}$ iterations

Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

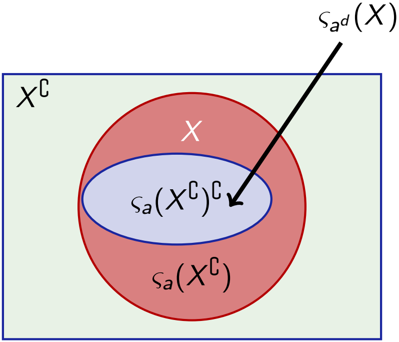
$$\varsigma_{a^d}(X) = (\varsigma_a(X^\complement))^\complement$$



Differential Game Logic: Denotational Semantics

Definition (Hybrid game *a*: denotational semantics)

$$\varsigma_{a^d}(X) = (\varsigma_a(X^\complement))^\complement$$


$\varsigma_{a^d}(X)$
$X^\complement$
X
$\varsigma_a(X^\complement)^\complement$
$\varsigma_a(X^\complement)$

Consistency & Determinacy

Theorem (Consistency & determinacy)

Hybrid games are consistent and determined, i.e. $\vDash \neg\langle a\rangle\neg P \leftrightarrow [a]P$.

Corollary (Determinacy: At least one player wins)

$\vDash \neg\langle a\rangle\neg P \rightarrow [a]P$, *thus* $\vDash \langle a\rangle\neg P \lor [a]P$.

Corollary (Consistency: At most one player wins)

$\vDash [a]P \rightarrow \neg\langle a\rangle\neg P$, *thus* $\vDash \neg([a]P \land \langle a\rangle\neg P)$

Outline

1. CPS Applications
2. Differential Game Logic
   - Differential Hybrid Games
   - Denotational Semantics
   - Determinacy
3. Proofs for CPS
   - Axiomatization
   - Soundness and Completeness
   - Corollaries
   - Separating Axioms
4. Expressiveness
5. Summary

Differential Game Logic: Axiomatization

$$[\cdot] \quad [a]P \leftrightarrow \neg\langle a\rangle\neg P$$

$$\langle:=\rangle \quad \langle x := f(x)\rangle p(x) \leftrightarrow p(f(x))$$

$$\langle'\rangle \quad \langle x' = f(x)\rangle P \leftrightarrow \exists t{\geq}0\, \langle x := y(t)\rangle P$$

$$\langle ?\rangle \quad \langle ?Q\rangle P \leftrightarrow (Q \land P)$$

$$\langle\cup\rangle \quad \langle a \cup b\rangle P \leftrightarrow \langle a\rangle P \lor \langle b\rangle P$$

$$\langle ;\rangle \quad \langle a; b\rangle P \leftrightarrow \langle a\rangle\langle b\rangle P$$

$$\langle^*\rangle \quad P \lor \langle a\rangle\langle a^*\rangle P \rightarrow \langle a^*\rangle P$$

$$\langle^d\rangle \quad \langle a^d\rangle P \leftrightarrow \neg\langle a\rangle\neg P$$

$$\text{M} \quad \frac{P \rightarrow Q}{\langle a\rangle P \rightarrow \langle a\rangle Q}$$

$$\text{FP} \quad \frac{P \lor \langle a\rangle Q \rightarrow Q}{\langle a^*\rangle P \rightarrow Q}$$

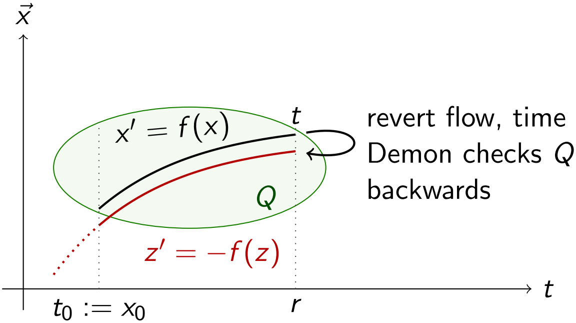
$$\text{MP} \quad \frac{P \quad P \rightarrow Q}{Q}$$

$$\forall \quad \frac{p \rightarrow Q}{p \rightarrow \forall x\, Q} \qquad (x \notin \text{FV}(p))$$

$$\text{US} \quad \frac{\varphi}{\varphi_{p(\cdot)}^{Q(\cdot)}}$$

“There and Back Again” Game

$$x' = f(x)\,\&\,Q \equiv t_0 := x_0; x' = f(x); (z := x; z' = -f(z))^d; ?(z_0 \geq t_0 \to Q(z))$$

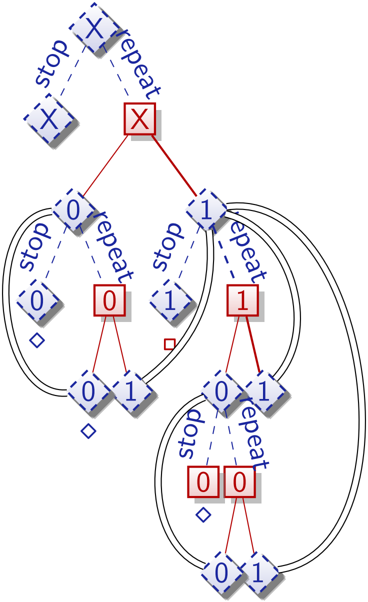


revert flow, time x_0;
Demon checks Q
backwards

Lemma

Evolution domains definable by games

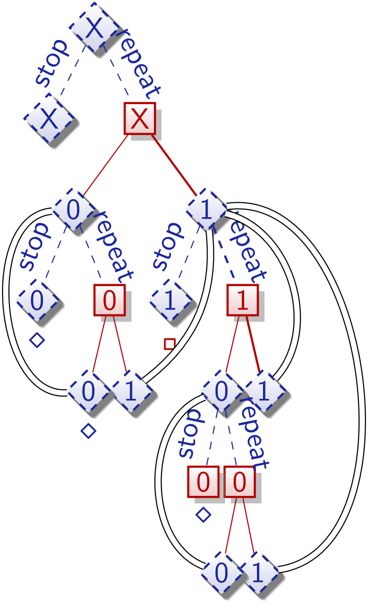
Example Proof: Dual Filibuster



$$\langle{}^d\rangle\ \frac{}{x = 0 \rightarrow \langle (x := 0 \cup x := 1)^\times \rangle x = 0}$$

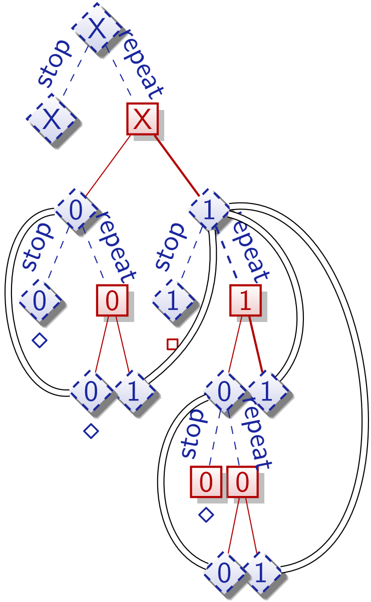
Example Proof: Dual Filibuster

$$\text{ind}\ \dfrac{\langle^d\rangle\ \dfrac{}{x = 0 \rightarrow [(x := 0 \cap x := 1)^*]x = 0}}{x = 0 \rightarrow \langle (x := 0 \cup x := 1)^\times \rangle x = 0}$$



Example Proof: Dual Filibuster

$$\dfrac{\dfrac{\dfrac{}{x=0 \to [x:=0 \cap x:=1]x=0}\,[\cdot]}{x=0 \to [(x:=0 \cap x:=1)^*]x=0}\,\text{ind}}{x=0 \to \langle (x:=0 \cup x:=1)^{\times} \rangle x=0}\,\langle^d\rangle$$



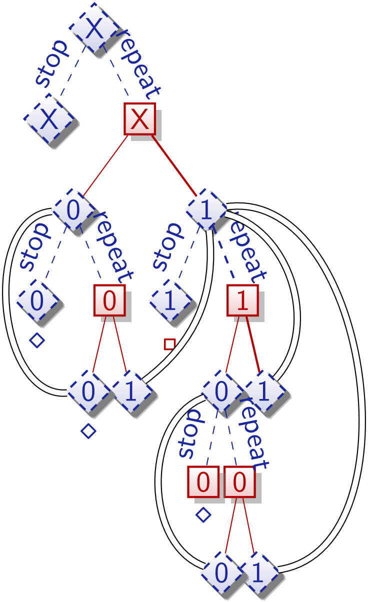
Example Proof: Dual Filibuster

$$\langle{}^d\rangle\;\overline{x = 0 \to \neg\langle x := 0 \cap x := 1\rangle \neg x = 0}$$

$$[\cdot]\;\overline{x = 0 \to [x := 0 \cap x := 1]x = 0}$$

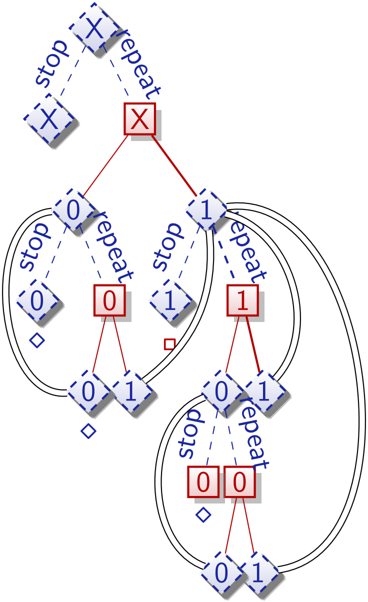
$$\text{ind}\;\overline{x = 0 \to [(x := 0 \cap x := 1)^*]x = 0}$$

$$\langle{}^d\rangle\;\overline{x = 0 \to \langle (x := 0 \cup x := 1)^{\times}\rangle x = 0}$$



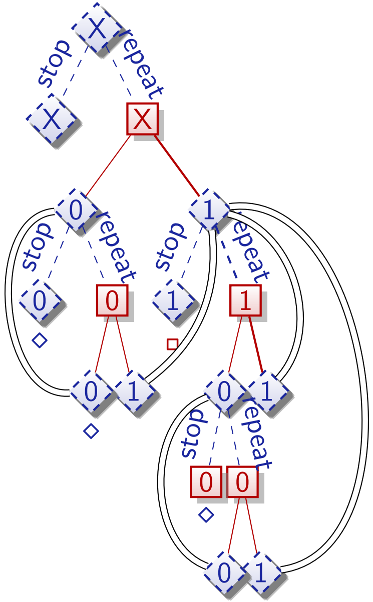
Example Proof: Dual Filibuster

$$\begin{array}{rl}
\langle\cup\rangle & \overline{x=0 \rightarrow \langle x:=0 \cup x:=1\rangle x=0} \\
\langle^d\rangle & \overline{x=0 \rightarrow \neg\langle x:=0 \cap x:=1\rangle \neg x=0} \\
[\cdot] & \overline{x=0 \rightarrow [x:=0 \cap x:=1]x=0} \\
\text{ind} & \overline{x=0 \rightarrow [(x:=0 \cap x:=1)^*]x=0} \\
\langle^d\rangle & \overline{x=0 \rightarrow \langle(x:=0 \cup x:=1)^\times\rangle x=0}
\end{array}$$



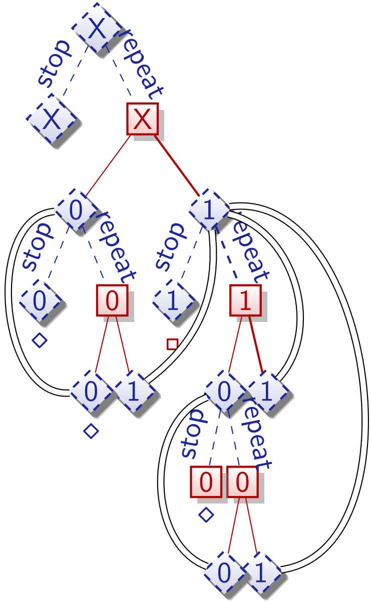
Example Proof: Dual Filibuster

$$\begin{array}{ll}
\langle:=\rangle & \\ \hline
 & x=0 \rightarrow \langle x:=0\rangle x=0 \lor \langle x:=1\rangle x=0 \\
\langle\cup\rangle & \\ \hline
 & x=0 \rightarrow \langle x:=0 \cup x:=1\rangle x=0 \\
\langle^d\rangle & \\ \hline
 & x=0 \rightarrow \neg\langle x:=0 \cap x:=1\rangle \neg x=0 \\
[\cdot] & \\ \hline
 & x=0 \rightarrow [x:=0 \cap x:=1]x=0 \\
\text{ind} & \\ \hline
 & x=0 \rightarrow [(x:=0 \cap x:=1)^*]x=0 \\
\langle^d\rangle & \\ \hline
 & x=0 \rightarrow \langle (x:=0 \cup x:=1)^\times\rangle x=0
\end{array}$$



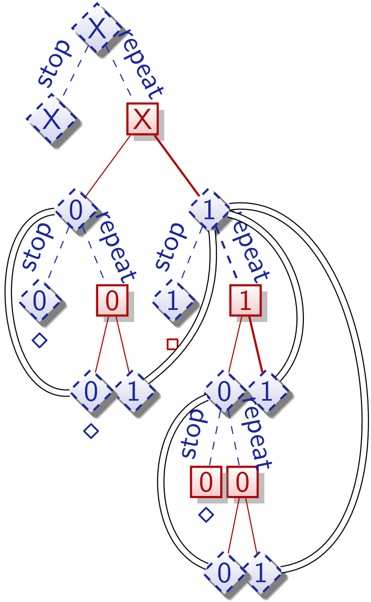
Example Proof: Dual Filibuster

$$\begin{array}{rl}
\mathbb{R} & \overline{x=0 \to 0=0 \lor 1=0} \\
\langle := \rangle & \overline{x=0 \to \langle x:=0\rangle x=0 \lor \langle x:=1\rangle x=0} \\
\langle \cup \rangle & \overline{x=0 \to \langle x:=0 \cup x:=1\rangle x=0} \\
\langle ^d \rangle & \overline{x=0 \to \neg\langle x:=0 \cap x:=1\rangle \neg x=0} \\
[\cdot] & \overline{x=0 \to [x:=0 \cap x:=1] x=0} \\
\text{ind} & \overline{x=0 \to [(x:=0 \cap x:=1)^*] x=0} \\
\langle ^d \rangle & \overline{x=0 \to \langle (x:=0 \cup x:=1)^\times \rangle x=0}
\end{array}$$



Example Proof: Dual Filibuster

$$\begin{array}{rl}
 & \qquad\qquad * \\ \cline{2-2}
\mathbb{R} & x=0 \to 0=0 \lor 1=0 \\ \cline{2-2}
\langle:=\rangle & x=0 \to \langle x:=0\rangle x=0 \lor \langle x:=1\rangle x=0 \\ \cline{2-2}
\langle\cup\rangle & x=0 \to \langle x:=0 \cup x:=1\rangle x=0 \\ \cline{2-2}
\langle^d\rangle & x=0 \to \neg\langle x:=0 \cap x:=1\rangle \neg x=0 \\ \cline{2-2}
[\cdot] & x=0 \to [x:=0 \cap x:=1]x=0 \\ \cline{2-2}
\text{ind} & x=0 \to [(x:=0 \cap x:=1)^*]x=0 \\ \cline{2-2}
\langle^d\rangle & x=0 \to \langle (x:=0 \cup x:=1)^{\times}\rangle x=0
\end{array}$$



Soundness & Completeness

Theorem (Completeness)

dG$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid games relative to any (differentially) expressive logic L.*

$$\vDash \varphi \quad \textit{iff} \quad \textit{Taut}_L \vdash \varphi$$

TOCL'15

Soundness & Completeness: Consequences

Corollary (Constructive)

Constructive and Moschovakis-coding-free. (Minimal: $x' = f(x)$, $\exists$, $[a^*]$)

Remark (Coquand & Huet) (Inf.Comput'88)

Modal analogue for $\langle a^* \rangle$ of characterizations in Calculus of Constructions

Corollary (Meyer & Halpern) (J.ACM'82)

$F \rightarrow \langle a \rangle G$ semidecidable for uninterpreted programs.

Corollary (Schmitt) (Inf.Control.'84)

$[a]$-free semidecidable for uninterpreted programs.

Corollary

Uninterpreted game logic with even ^{d} in $\langle a \rangle$ is semidecidable.

Soundness & Completeness: Consequences

Corollary

Harel'77 convergence rule unnecessary for hybrid games, hybrid systems, discrete programs.

Corollary (Characterization of hybrid game challenges)

- $[a^*]G$: *Succinct invariants* — *discrete* Π^0_2
- $[x' = f(x)]G$ *and* $\langle x' = f(x)\rangle G$: *Succinct differential (in)variants* — Δ^1_1
- $\exists x\, G$: *Complexity depends on Herbrand disjunctions:* — *discrete* Π^1_1

  ✓ *uninterpreted* ✓ *reals* × $\exists x\,[a^*]G$ Π^1_1*-complete for discrete* a

Corollary (Hybrid version of Parikh's result) (FOCS'83)

**-free* d$\mathcal{GL}$ *complete relative to* d$\mathcal{L}$*, relative to continuous, or to discrete*

^{d}*-free* d$\mathcal{GL}$ *complete relative to* d$\mathcal{L}$*, relative to continuous, or to discrete*

Soundness & Completeness: Consequences

Corollary (ODE Completeness) (+LICS'12)

dG$\mathcal{L}$ *complete relative to ODE for hybrid games with finite-rank Borel winning regions.*

Corollary (Continuous Completeness)

dG$\mathcal{L}$ *complete relative to* $L_{\mu D}$*, continuous modal* μ*, over* $\mathbb{R}$

Corollary (Discrete Completeness) (+LICS'12)

dG$\mathcal{L}$ $+$ *Euler axiom complete relative to discrete* L_{μ} *over* $\mathbb{R}$

Soundness & Completeness: Consequences

$$\langle(\underbrace{\underbrace{x := 1; x' = 1^d}_{b} \cup \underbrace{x := x - 1}_{c}}_{a})^*\rangle 0 \leq x < 1$$

▸ Fixpoint style proof technique

$$\begin{array}{r}
\hline
\forall x\, (0{\leq}x{<}1 \lor \forall t{\geq}0\, p(0 + t) \lor p(x - 1) \rightarrow p(x)) \rightarrow (\mathit{true} \rightarrow p(x)) \\
\hline
\forall x\, (0{\leq}x{<}1 \lor \langle x := 1\rangle\neg\exists t{\geq}0\, \langle x := x{+}t\rangle\neg p(x) \lor p(x{-}1) \rightarrow p(x)) \rightarrow (\mathit{true} \rightarrow p(x)) \\
\hline
\forall x\, (0{\leq}x{<}1 \lor \langle x := 1\rangle\neg\langle x' = 1\rangle\neg p(x) \lor p(x - 1) \rightarrow p(x)) \rightarrow (\mathit{true} \rightarrow p(x)) \\
\hline
\forall x\, (0{\leq}x{<}1 \lor \langle b\rangle p(x) \lor \langle c\rangle p(x) \rightarrow p(x)) \rightarrow (\mathit{true} \rightarrow p(x)) \\
\hline
\forall x\, (0{\leq}x{<}1 \lor \langle b \cup c\rangle p(x) \rightarrow p(x)) \rightarrow (\mathit{true} \rightarrow p(x)) \\
\hline
\forall x\, (0{\leq}x{<}1 \lor \langle a\rangle\langle a^*\rangle 0{\leq}x{<}1 \rightarrow \langle a^*\rangle 0{\leq}x{<}1) \rightarrow (\mathit{true} \rightarrow \langle a^*\rangle 0{\leq}x{<}1) \\
\hline
\mathit{true} \rightarrow \langle a^*\rangle 0{\leq}x{<}1
\end{array}$$

Separating Axioms

Theorem (Axiomatic separation: hybrid systems vs. hybrid games)

Axiomatic separation is exactly K, I, C, B, V, G. dG$\mathcal{L}$ *is a subregular, sub-Barcan, monotonic modal logic without loop induction axioms.*

$\cancel{\mathrm{K}}\quad [a](P \to Q) \to ([a]P \to [a]Q)$ $\qquad$ $\mathrm{M}_{[\cdot]}\quad \dfrac{P \to Q}{[a]P \to [a]Q}$

$\cancel{\overleftarrow{\mathrm{M}}}\quad \langle a\rangle(P \lor Q) \to \langle a\rangle P \lor \langle a\rangle Q$ $\qquad$ $\mathrm{M}\quad \langle a\rangle P \lor \langle a\rangle Q \to \langle a\rangle(P \lor Q)$

$\cancel{\mathrm{I}}\quad [a^*](P \to [a]P) \to (P \to [a^*]P)$ $\qquad$ $\forall\mathrm{I}\quad (P \to [a]P) \to (P \to [a^*]P)$

$$\cancel{\mathrm{C}}\quad \begin{array}{l}[a^*]\forall v{>}0\,(p(v) \to \langle a\rangle p(v-1))\\ \quad\to \forall v\,(p(v) \to \langle a^*\rangle \exists v{\leq}0\, p(v))\end{array}\quad (v\notin a)$$

$\cancel{\mathrm{B}}\quad \langle a\rangle\exists x\, P \to \exists x\, \langle a\rangle P \qquad (x\notin a)$ $\qquad$ $\overleftarrow{\mathrm{B}}\quad \exists x\, \langle a\rangle P \to \langle a\rangle \exists x\, P$

$\cancel{\mathrm{V}}\quad p \to [a]p \qquad (\mathrm{FV}(p) \cap \mathrm{BV}(a) = \emptyset)$ $\qquad$ $\mathrm{VK}\quad p \to ([a]\mathit{true} \to [a]p)$

$\cancel{\mathrm{G}}\quad \dfrac{P}{[a]P}$ $\qquad$ $\mathrm{M}_{[\cdot]}\quad \dfrac{P \to Q}{[a]P \to [a]Q}$

Outline

1. CPS Applications
2. Differential Game Logic
   - Differential Hybrid Games
   - Denotational Semantics
   - Determinacy
3. Proofs for CPS
   - Axiomatization
   - Soundness and Completeness
   - Corollaries
   - Separating Axioms
4. Expressiveness
5. Summary

Expressiveness

Theorem (Expressive Power: hybrid systems $<$ hybrid games)

dG$\mathcal{L}$ *for hybrid games strictly more expressive than* d$\mathcal{L}$ *for hybrid games:*

$$\mathrm{d}\mathcal{L} < \mathrm{dG}\mathcal{L}$$

Expressiveness

Theorem (Expressive Power: hybrid systems < hybrid games)

dG$\mathcal{L}$ *for hybrid games strictly more expressive than* d$\mathcal{L}$ *for hybrid games:*

$$\mathrm{d}\mathcal{L} < \mathrm{dG}\mathcal{L}$$

First-order adm. $\mathbb{R}$

Inductive adm. $\mathbb{R}$

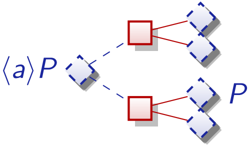
Outline

1. CPS Applications
2. Differential Game Logic
    - Differential Hybrid Games
    - Denotational Semantics
    - Determinacy
3. Proofs for CPS
    - Axiomatization
    - Soundness and Completeness
    - Corollaries
    - Separating Axioms
4. Expressiveness
5. Summary

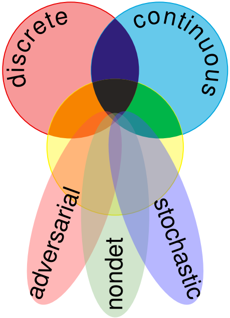
Differential Game Logic

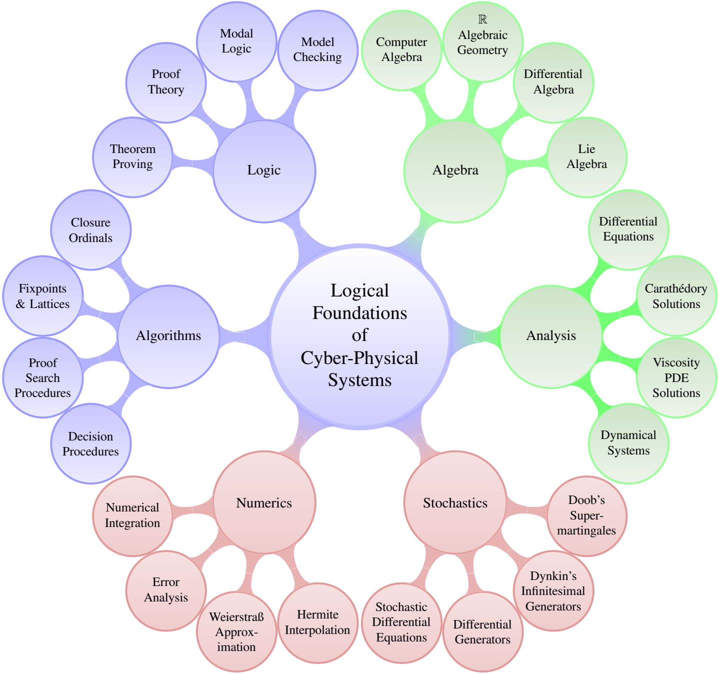
differential game logic

$d\mathrm{G}\mathcal{L} = \mathrm{GL} + \mathrm{HG} = d\mathcal{L} + {}^{d}$



- Logic for hybrid games
- Compositional PL + logic
- Discrete + continuous + adversarial
- Winning region iteration $\geq\omega_1^{\mathrm{CK}}$
- Sound & rel. complete axiomatization
- Hybrid games > hybrid systems
- ^{d} radical challenge yet smooth extension
- Stochastic ≈ adversarial




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

André Platzer.
Differential game logic.
ACM Trans. Comput. Log., 2015.
To appear. Preprint at arXiv 1408.1980.
`doi:10.1145/2817824`.

André Platzer.
Differential hybrid games.
CoRR, abs/1507.04943, 2015.
`arXiv:1507.04943`.

André Platzer.
Logics of dynamical systems.
In LICS [9], pages 13–24.
`doi:10.1109/LICS.2012.13`.

André Platzer.
The complete proof theory of hybrid systems.
In LICS [9], pages 541–550.
`doi:10.1109/LICS.2012.64`.

André Platzer.
Differential game logic for hybrid games.
Technical Report CMU-CS-12-105, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, March 2012.

Jan-David Quesel and André Platzer.
Playing hybrid games with KeYmaera.
In Bernhard Gramlich, Dale Miller, and Ulrike Sattler, editors, *IJCAR*, volume 7364 of *LNCS*, pages 439–453. Springer, 2012.
doi:10.1007/978-3-642-31365-3_34.

André Platzer.
A complete axiomatization of differential game logic for hybrid games.
Technical Report CMU-CS-13-100R, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, January, Revised and extended in July 2013.

André Platzer.
Differential game logic.
CoRR, abs/1408.1980, 2014.

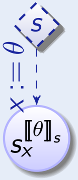
arXiv:1408.1980.

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

Differential Game Logic: Operational Semantics

Definition (Hybrid game *a*: operational semantics)

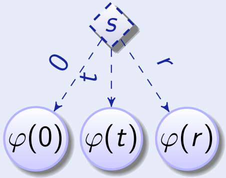
$x := \theta$



Differential Game Logic: Operational Semantics

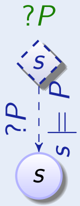
Definition (Hybrid game *a*: operational semantics)

$x' = \theta \,\&\, Q$



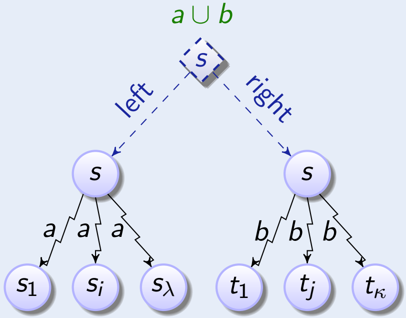
Differential Game Logic: Operational Semantics

Definition (Hybrid game a: operational semantics)



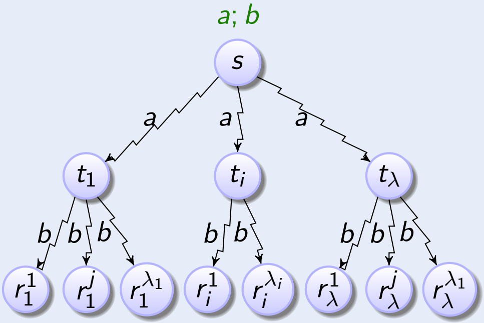
Differential Game Logic: Operational Semantics

Definition (Hybrid game a: operational semantics)



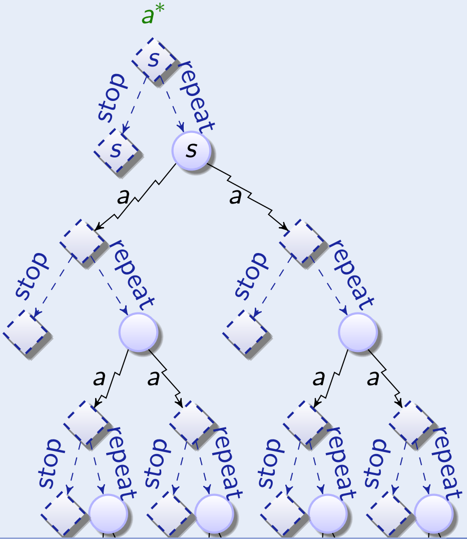
Differential Game Logic: Operational Semantics

Definition (Hybrid game *a*: operational semantics)



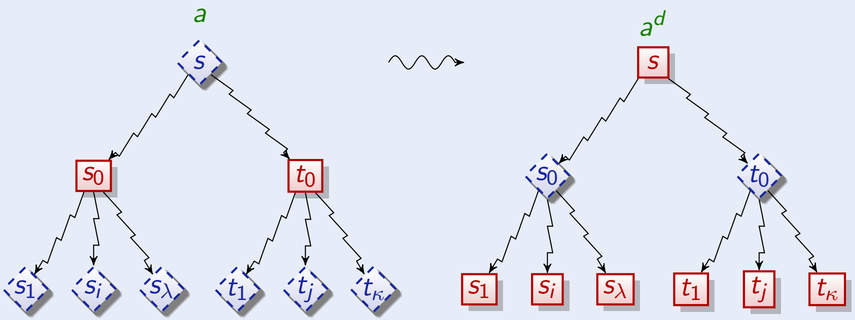
Differential Game Logic: Operational Semantics

Definition (Hybrid game a: operational semantics)



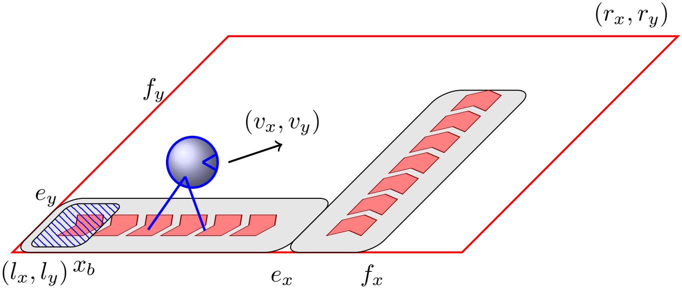
Differential Game Logic: Operational Semantics

Definition (Hybrid game a: operational semantics)



Successful Hybrid Games Proofs

Verification Challenge:



Hybrid games proving also for proving relaxed notions of system similarity

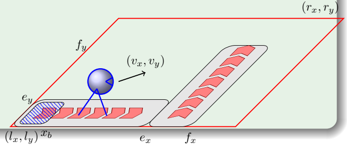
Robotic Factory Automation (*RF*)

Example (Environment vs. Robot)

$$\Big(\big(\,?\mathit{true} \cap (?(x < e_x \land y < e_y \land \mathrm{eff}_1 = 1);\ v_x := v_x + c_x;\ \mathrm{eff}_1 := 0)$$

$$\cap\, (?(e_x \le x \land y \le f_y \land \mathrm{eff}_2 = 1);\ v_y := v_y + c_y;\ \mathrm{eff}_2 := 0)\ \big)\,;$$

$$\Big)^{\times}$$



Robotic Factory Automation (*RF*)

Example (Environment vs. Robot)

$$\Big(\big(\,?true \cap (?(x < e_x \land y < e_y \land \mathrm{eff}_1 = 1);\ v_x := v_x + c_x;\ \mathrm{eff}_1 := 0)$$

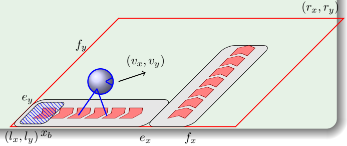
$$\cap\, (?(e_x \le x \land y \le f_y \land \mathrm{eff}_2 = 1);\ v_y := v_y + c_y;\ \mathrm{eff}_2 := 0)\ \big)\,;$$

$$(\,a_x := *;\ ?(-A \le a_x \le A);$$

$$a_y := *;\ ?(-A \le a_y \le A);$$

$$t_s := 0\ )\,;$$

$$\Big)^{\times}$$



Robotic Factory Automation (RF)

Example (Environment vs. Robot)

$$\Big(\big(\,?true \cap \big(?(x < e_x \land y < e_y \land \mathrm{eff}_1 = 1);\ v_x := v_x + c_x;\ \mathrm{eff}_1 := 0\big)$$

$$\cap \big(?(e_x \le x \land y \le f_y \land \mathrm{eff}_2 = 1);\ v_y := v_y + c_y;\ \mathrm{eff}_2 := 0\big)\,\big)\,;$$

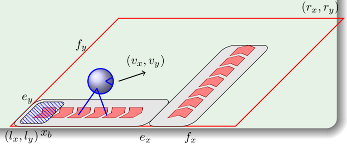
$$(\,a_x := *;\ ?(-A \le a_x \le A);$$

$$a_y := *;\ ?(-A \le a_y \le A);$$

$$t_s := 0\,)\,;$$

$$(\,x' = v_x, y' = v_y, v_x' = a_x, v_y' = a_y, t' = 1, t_s' = 1 \,\&\, t_s \le \varepsilon\,)^d\,;$$

$$\Big)^{\times}$$



Robotic Factory Automation (*RF*)

Example (Environment vs. Robot)

$$\Big(\big(?true \cap (?(x < e_x \land y < e_y \land \mathrm{eff}_1 = 1);\ v_x := v_x + c_x;\ \mathrm{eff}_1 := 0)$$

$$\cap\, (?(e_x \le x \land y \le f_y \land \mathrm{eff}_2 = 1);\ v_y := v_y + c_y;\ \mathrm{eff}_2 := 0)\ \big);$$

$$(\,a_x := *;\ ?(-A \le a_x \le A);$$

$$a_y := *;\ ?(-A \le a_y \le A);$$

$$t_s := 0\ );$$

$$\big(\,(x' = v_x, y' = v_y, v_x' = a_x, v_y' = a_y, t' = 1, t_s' = 1 \& t_s \le \varepsilon\,)^d\,;$$

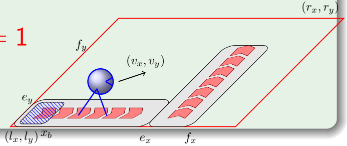
$$\cup\,((\,?a_x v_x \le 0 \land a_y v_y \le 0;$$

$$\text{if } v_x = 0 \text{ then } a_x := 0 \text{ fi};$$

$$\text{if } v_y = 0 \text{ then } a_y := 0 \text{ fi}\ );$$

$$(x' = v_x, y' = v_y, v_x' = a_x, v_y' = a_y, t' = 1, t_s' = 1$$

$$\& t_s \le \varepsilon \land a_x v_x \le 0 \land a_y v_y \le 0)^d))\Big)^\times$$



Robotic Factory Automation (*RF*)

Proposition (Robot stays in □)

$$\vDash (x = y = 0 \land v_x = v_y = 0 \land \text{▸ Controllability Assumptions}) \\ \to (RF)(x \in [l_x, r_x] \land y \in [l_y, r_y])$$

Proposition (Stays in □ + leaves shaded region in time)

$RF|_x$: RF projected to the x-axis

$$\vDash (x = 0 \land v_x = 0 \land \text{▸ Controllability Assumptions}) \\ \to (RF|_x)(x \in [l_x, r_x] \land (t \geq \varepsilon \to (x \geq x_b)))$$