Uniform Substitution At One Fell Swoop

André Platzer

Carnegie Mellon University

In Shakespeare's 1611 play, "*at one fell swoop*" was likened to the suddenness with which a bird of prey fiercely attacks a whole nest at once.

Outline

Outline

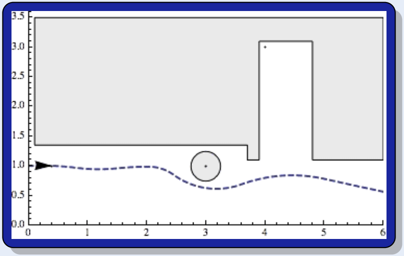
1. Motivation
   - Parsimonious Hybrid Game Proofs
   - Foundation for Verification
2. Differential Game Logic
   - Syntax
   - Example: Push-around Cart
   - Denotational Semantics
3. Uniform Substitution
   - Application
   - Uniform Substitution Lemma
   - Uniform Substitution of Rules
   - Static Semantics
   - Axioms
   - Differential Hybrid Games
4. Summary

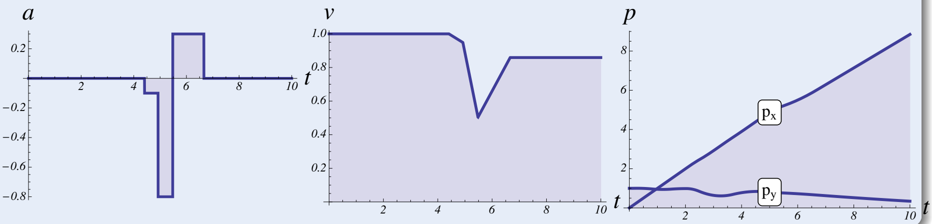
CPS Analysis: Robot Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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



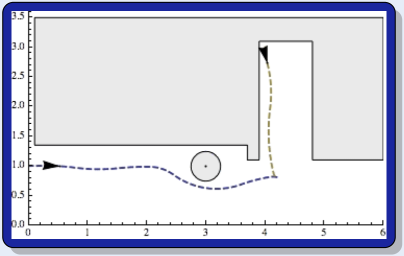


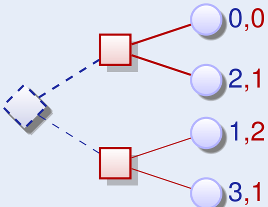
CPS Analysis: Robot Control

Challenge (Games)

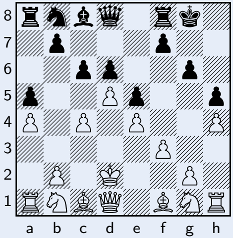
Game rules describing play evolution with both

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





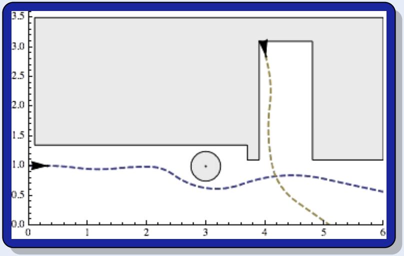
◇\□	Tr	Pl
Trash	1,2	0,0
Plant	0,0	2,1

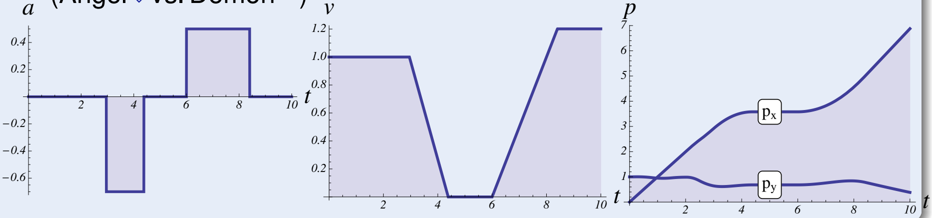


Challenge (Hybrid Games)

Game rules describing play evolution with

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)
- Adversarial dynamics (Angel $\diamond$ vs. Demon $\square$)

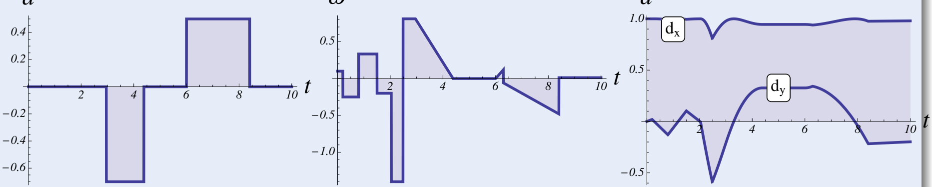




Challenge (Hybrid Games)

Game rules describing play evolution with

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



Foundation for Verification

Foundation for ↑

FOL	Functional Language	Imperative Language
Formula	Functional program	Imperative program/game
Predicate calculus	Function calculus	Program calculus
Subst + rename	α, β, η-conversion	USubst + rename

Functional

α-conversion	for bound variables
β-reduction	capture-avoiding subst.
η-conversion	versus free variables

Imperative

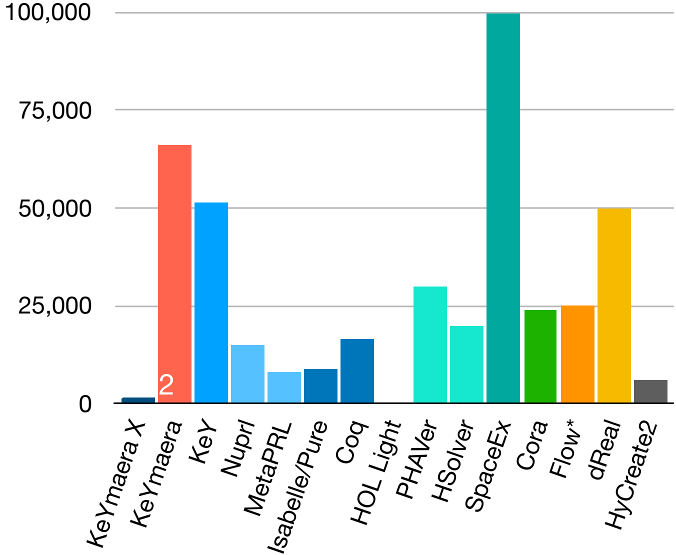
Uniform substitution replaces predicate/function/program sym. mindful of free/bound variables

Substitution is fundamental but subtle. Henkin wants it banished!

Now: Make USubst even more subtle, but faster, and still sound.

Beware: Imperative free and bound variables may overlap!

KeYmaera X Microkernel for Soundness 1 700 LOC

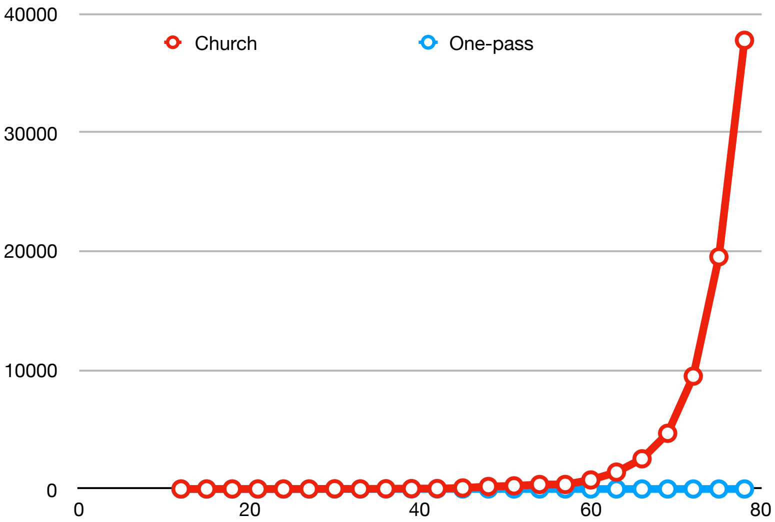


Games: months ↘ minutes

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

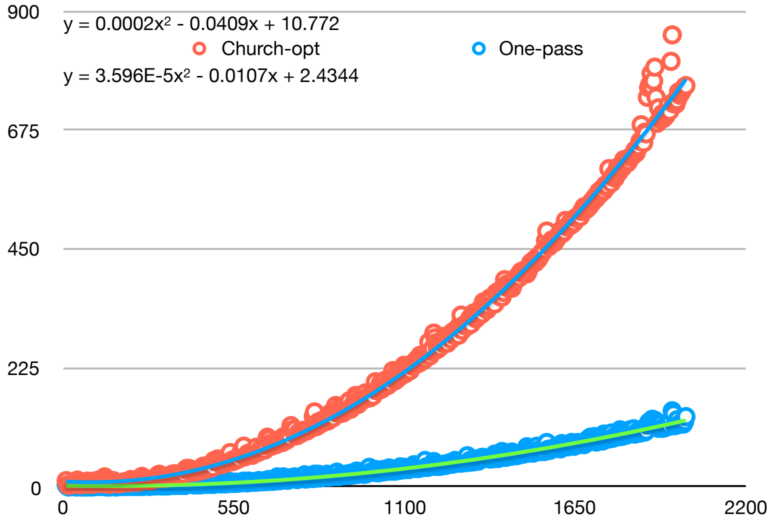
Experiments

Church checks exponentially (sometimes & in unoptimized implementations)



Experiments

Church checks quadratically (invasive space-time tradeoff optimizations)



Outline

Differential Game Logic: Syntax

Definition (Hybrid game α)

$$a \mid x := \theta \mid ?q \mid x' = \theta \mid \alpha \cup \beta \mid \alpha ; \beta \mid \alpha^* \mid \alpha^d$$

Definition (dGL Formula ϕ)

$$p(\theta_1, \dots, \theta_n) \mid \theta \geq \eta \mid \neg\phi \mid \phi \land \psi \mid \forall x\, \phi \mid \exists x\, \phi \mid \langle\alpha\rangle\phi \mid [\alpha]\phi$$

TOCL'15

Differential Game Logic: Syntax

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

Definition (Hybrid game α)

$$a \mid x := \theta \mid ?q \mid x' = \theta \mid \alpha \cup \beta \mid \alpha ; \beta \mid \alpha^* \mid \alpha^d$$

Definition (dGL Formula ϕ)

$$p(\theta_1, \ldots, \theta_n) \mid \theta \geq \eta \mid \neg\phi \mid \phi \wedge \psi \mid \forall x\, \phi \mid \exists x\, \phi \mid \langle\alpha\rangle\phi \mid [\alpha]\phi$$

All Reals | Some Reals

TOCL'15

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

Definition (Hybrid game α)

$$a \mid x := \theta \mid ?q \mid x' = \theta \mid \alpha \cup \beta \mid \alpha;\beta \mid \alpha^* \mid \alpha^d$$

Definition (dGL Formula ϕ)

$$p(\theta_1,\dots,\theta_n) \mid \theta \geq \eta \mid \neg\phi \mid \phi \wedge \psi \mid \forall x\, \phi \mid \exists x\, \phi \mid \langle\alpha\rangle\phi \mid [\alpha]\phi$$

All Reals | Some Reals

TOCL'15

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

Definition (Hybrid game α)

$$a \mid x := \theta \mid ?q \mid x' = \theta \mid \alpha \cup \beta \mid \alpha;\beta \mid \alpha^* \mid \alpha^d$$

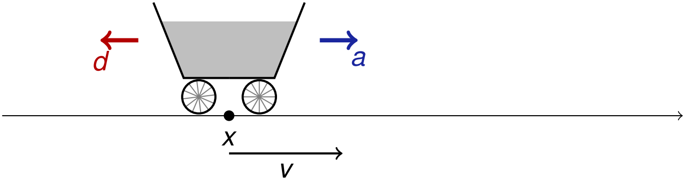
Definition (dGL Formula ϕ)

$$p(\theta_1, \ldots, \theta_n) \mid \theta \geq \eta \mid \neg\phi \mid \phi \wedge \psi \mid \forall x\, \phi \mid \exists x\, \phi \mid \langle\alpha\rangle\phi \mid [\alpha]\phi$$

All Reals | Some Reals | Angel Wins | Demon Wins

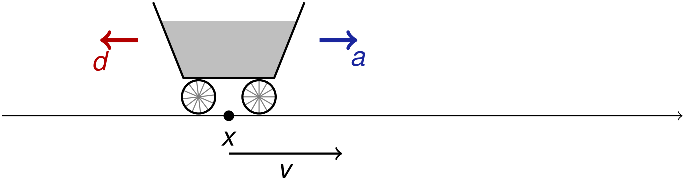
TOCL'15

Example: Push-around Cart



$$v \geq 1 \rightarrow$$

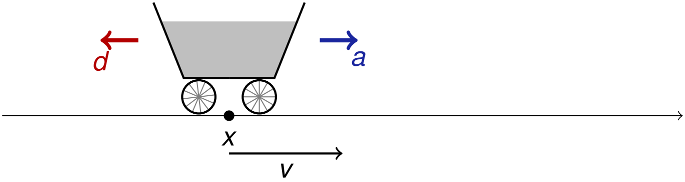
$$\left[\left((d := 1 \cup d := -1)^d; (a := 1 \cup a := -1); \{x' = v, v' = a + d\}\right)^*\right] v \geq 0$$



$\vDash v \geq 1 \rightarrow$ *d* before *a* can compensate

$$[((d := 1 \cap d := -1); (a := 1 \cup a := -1); \{x' = v, v' = a + d\})^*]\, v \geq 0$$

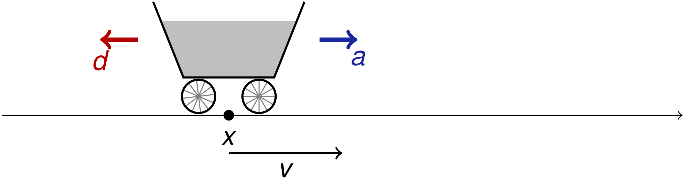
Example: Push-around Cart



$\vDash v \geq 1 \rightarrow$ *d* before *a* can compensate

$$[((d := 1 \cap d := -1); (a := 1 \cup a := -1); \{x' = v, v' = a + d\})^*] \, v \geq 0$$

$$\langle ((d := 1 \cap d := -1); (a := 1 \cup a := -1); \\ t := 0; \{x' = v, v' = a + d, t' = 1 \,\&\, t \leq 1\})^* \rangle \, x^2 \geq 100$$



$\vDash v \geq 1 \rightarrow$ *d* before *a* can compensate

$$[((d := 1 \cap d := -1); (a := 1 \cup a := -1); \{x' = v, v' = a + d\})^*]\, v \geq 0$$

$\vDash \langle((d := 1 \cap d := -1); (a := 1 \cup a := -1);$ *a* := *d* then *a* := sign *v*

$$t := 0; \{x' = v, v' = a + d, t' = 1 \,\&\, t \leq 1\})^*\rangle\, x^2 \geq 100$$

Differential Game Logic: Denotational Semantics

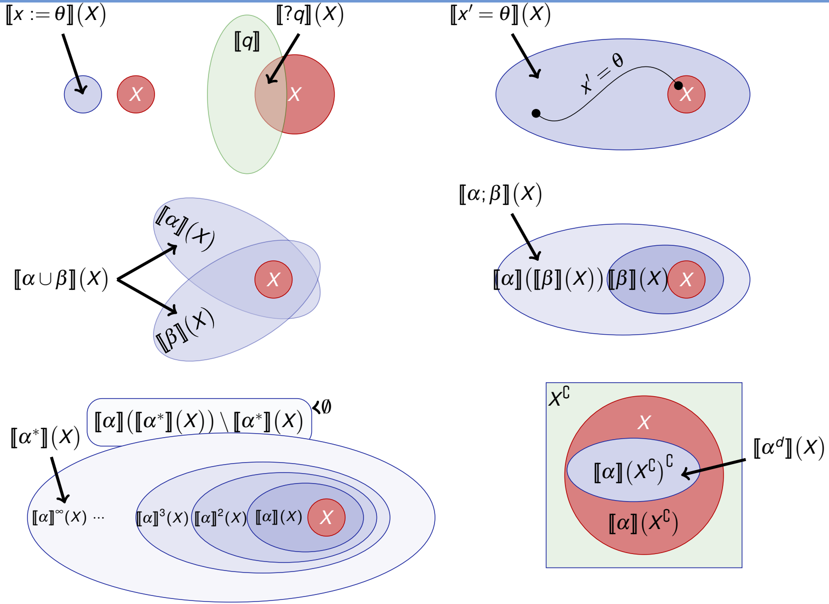
Definition (Hybrid game α) $[\![\cdot]\!] : \mathrm{HG} \to (\wp(\mathscr{S}) \to \wp(\mathscr{S}))$

$$\begin{aligned}
[\![x := \theta]\!](X) &= \{\omega \in \mathscr{S} : \omega_x^{\omega[\![\theta]\!]} \in X\} \\
[\![x' = \theta]\!](X) &= \{\varphi(0) \in \mathscr{S} : \varphi(r) \in X, \frac{\mathrm{d}\,\varphi(t)(x)}{\mathrm{d}t}(\zeta) = \varphi(\zeta)[\![\theta]\!] \text{ for all } \zeta\} \\
[\![?q]\!](X) &= [\![q]\!] \cap X \\
[\![\alpha \cup \beta]\!](X) &= [\![\alpha]\!](X) \cup [\![\beta]\!](X) \\
[\![\alpha;\beta]\!](X) &= [\![\alpha]\!]([\![\beta]\!](X)) \\
[\![\alpha^*]\!](X) &= \bigcap\{Z \subseteq \mathscr{S} : X \cup [\![\alpha]\!](Z) \subseteq Z\} \\
[\![\alpha^d]\!](X) &= ([\![\alpha]\!](X^\complement))^\complement
\end{aligned}$$

Definition (dGL Formula ϕ) $[\![\cdot]\!] : \mathrm{Fml} \to \wp(\mathscr{S})$

$$\begin{aligned}
[\![\theta \geq \eta]\!] &= \{\omega \in \mathscr{S} : \omega[\![\theta]\!] \geq \omega[\![\eta]\!]\} \\
[\![\neg\phi]\!] &= ([\![\phi]\!])^\complement \\
[\![\phi \wedge \psi]\!] &= [\![\phi]\!] \cap [\![\psi]\!] \\
[\![\langle\alpha\rangle\phi]\!] &= [\![\alpha]\!]([\![\phi]\!]) \\
[\![[\alpha]\phi]\!] &= [\![\alpha]\!]([\![\phi]\!]^\complement)^\complement
\end{aligned}$$

Differential Game Logic: Denotational Semantics



Outline

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 **no** operator $\otimes$
are free in the substitution on its argument θ (U-admissible)

$$\text{US}\ \frac{\langle a \cup b\rangle p(\bar{x}) \leftrightarrow \langle a\rangle p(\bar{x}) \lor \langle b\rangle p(\bar{x})}{\langle v := v+1 \cup x' = v\rangle x > 0 \leftrightarrow \langle v := v+1\rangle x > 0 \lor \langle x' = v\rangle x > 0}$$

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 **no** operator $\otimes$
are free in the substitution on its argument θ (U-admissible)

$$\frac{\langle v := f\rangle p(v) \leftrightarrow p(f)}{\langle v := -x\rangle\langle x' = v\rangle\, x \geq 0 \leftrightarrow \langle x' = -x\rangle\, x \geq 0}$$

Uniform Substitution

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

Modular interface: Prover vs. Logic

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

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 **no** operator $\otimes$
are free in the substitution on its argument θ (U-admissible)

If you bind a free variable, you go to logic jail!

$$\frac{\langle v := f\rangle p(v) \leftrightarrow p(f)}{\langle v := -x\rangle\langle x' = v\rangle\, x \geq 0 \leftrightarrow \langle x' = -x\rangle\, x \geq 0}$$

Clash

Uniform Substitution

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

Modular interface: Prover vs. Logic

$$US \quad \frac{\phi}{\sigma\phi}$$

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 **no** operator $\otimes$
are free in the substitution on its argument θ (U-admissible)

If you bind a free variable, you go to logic jail!

$$\frac{\langle x' = f(x), y' = a(x)y\rangle\, x \geq 1 \leftrightarrow \langle x' = f(x)\rangle\, x \geq 1}{\langle x' = x^2, y' = zy\, y\rangle\, x \geq 1 \leftrightarrow \langle x' = x^2\rangle\, x \geq 1}$$

Uniform Substitution

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

Modular interface: Prover vs. Logic

$$US \quad \frac{\phi}{\sigma\phi}$$

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

i.e. bound variables $U = \text{BV}(\otimes(\cdot))$ of **no** operator $\otimes$
are free in the substitution on its argument θ (U-admissible)

If you bind a free variable, you go to logic jail!

$$\frac{\langle x' = f(x), y' = a(x)y\rangle\, x \geq 1 \leftrightarrow \langle x' = f(x)\rangle\, x \geq 1}{\langle x' = x^2, y' = zy\, y\rangle\, x \geq 1 \leftrightarrow \langle x' = x^2\rangle\, x \geq 1}$$

Clash

Uniform Substitution Application: Church-style

$$
\begin{array}{rcll}
\sigma(f(\theta)) &=& (\sigma f)(\sigma\theta) & \\
&\overset{\text{def}}{=}& \{\cdot \mapsto \sigma\theta\}\sigma f(\cdot) & \\
\sigma(\theta+\eta) &=& \sigma\theta + \sigma\eta & \\
\sigma((\theta)') &=& (\sigma\theta)' & \text{if } \sigma\ \mathbb{V}\text{-admissible for } \theta \\
\hline
\sigma(p(\theta)) &=& (\sigma p)(\sigma\theta) & \\
\sigma(\phi \land \psi) &=& \sigma\phi \land \sigma\psi & \\
\sigma(\forall x\,\phi) &=& \forall x\,\sigma\phi & \text{if } \sigma\ \{x\}\text{-admissible for } \phi \\
\sigma(\langle\alpha\rangle\phi) &=& \langle\sigma\alpha\rangle\sigma\phi & \text{if } \sigma\ \mathrm{BV}(\sigma\alpha)\text{-admissible for } \phi \\
\hline
\sigma(a) &=& \sigma a & \\
\sigma(x:=\theta) &=& x:=\sigma\theta & \\
\sigma(x'=\theta\,\&\,q) &=& x'=\sigma\theta\,\&\,\sigma q & \text{if } \sigma\ \{x,x'\}\text{-admissible for } \theta, q \\
\sigma(\alpha\cup\beta) &=& \sigma\alpha\cup\sigma\beta & \\
\sigma(\alpha;\beta) &=& \sigma\alpha;\sigma\beta & \text{if } \sigma\ \mathrm{BV}(\sigma\alpha)\text{-admissible for } \beta \\
\sigma(\alpha^*) &=& (\sigma\alpha)^* & \text{if } \sigma\ \mathrm{BV}(\sigma\alpha)\text{-admissible for } \alpha \\
\sigma(\alpha^d) &=& (\sigma\alpha)^d &
\end{array}
$$

Uniform Substitution Application: Church-style

$$\sigma(f(\theta)) = (\sigma f)(\sigma\theta)$$
$$\overset{\text{def}}{=} \{\cdot \mapsto \sigma\theta\}\sigma f(\cdot)$$
$$\sigma(\theta + \eta) = \sigma\theta + \sigma\eta$$
$$\sigma((\theta)') = (\sigma\theta)' \quad \text{if } \sigma\ \mathbb{V}\text{-admissible for } \theta$$

$$\sigma(p(\theta)) = (\sigma p)(\sigma\theta)$$
$$\sigma(\phi \wedge \psi) = \sigma\phi \wedge \sigma\psi$$
$$\sigma(\forall x\, \phi) = \forall x\, \sigma\phi \quad \text{if } \sigma\ \{x\}\text{-admissible for } \phi$$
$\sigma(\langle\alpha\rangle$ … ssible for ϕ

$\sigma($ …

$\sigma(x :=$ …

$$\sigma(x' = \theta \,\&\, q) = x' = \sigma\theta \,\&\, \sigma q \quad \text{if } \sigma\ \{x, x'\}\text{-admissible for } \theta, q$$
$$\sigma(\alpha \cup \beta) = \sigma\alpha \cup \sigma\beta$$
$$\sigma(\alpha;\beta) = \sigma\alpha;\sigma\beta \quad \text{if } \sigma\ \mathrm{BV}(\sigma\alpha)\text{-admissible for } \beta$$
$$\sigma(\alpha^*) = (\sigma\alpha)^* \quad \text{if } \sigma\ \mathrm{BV}(\sigma\alpha)\text{-admissible for } \alpha$$
$$\sigma(\alpha^d) = (\sigma\alpha)^d$$

Idea

Check side conditions at each operator again where soundness demands it.

Uniform Substitution Application: One-pass with Taboo U

$$
\begin{aligned}
\sigma^U(f(\theta)) &= (\sigma^U f)(\sigma^U\theta) \\
&\overset{\text{def}}{=} \{\cdot \mapsto \sigma^U\theta\}^{\emptyset}\sigma f(\cdot) && \text{if } FV(\sigma f(\cdot)) \cap U = \emptyset \\
\sigma^U(\theta+\eta) &= \sigma^U\theta + \sigma^U\eta \\
\sigma^U((\theta)') &= (\sigma^{\mathbb{V}}\theta)'
\end{aligned}
$$

$$
\begin{aligned}
\sigma^U(p(\theta)) &= (\sigma^U p)(\sigma^U\theta) && \text{if } FV(\sigma p(\cdot)) \cap U = \emptyset \\
\sigma^U(\phi \land \psi) &= \sigma^U\phi \land \sigma^U\psi \\
\sigma^U(\forall x\,\phi) &= \forall x\,\sigma^{U\cup\{x\}}\phi \\
\sigma^U(\langle\alpha\rangle\phi) &= \langle\sigma^U_V\alpha\rangle\sigma^V\phi
\end{aligned}
$$

$$
\begin{aligned}
\sigma^U_{U\cup BV(\sigma a)}(a) &= \sigma a \\
\sigma^U_{U\cup\{x\}}(x:=\theta) &= x:=\sigma^U\theta \\
\sigma^U_{U\cup\{x,x'\}}(x'=\theta\,\&\,q) &= (x'=\sigma^{U\cup\{x,x'\}}\theta\,\&\,\sigma^{U\cup\{x,x'\}}q) \\
\sigma^U_{V\cup W}(\alpha\cup\beta) &= \sigma^U_V\alpha \cup \sigma^U_W\beta \\
\sigma^U_W(\alpha;\beta) &= \sigma^U_V\alpha;\sigma^V_W\beta \\
\sigma^U_V(\alpha^*) &= (\sigma^V_V\alpha)^* && \text{where } \sigma^U_V\alpha \text{ defined} \\
\sigma^U_V(\alpha^d) &= (\sigma^U_V\alpha)^d
\end{aligned}
$$

$$
\begin{aligned}
\sigma^U(f(\theta)) &= (\sigma^U f)(\sigma^U \theta) \\
&\overset{\text{def}}{=} \{\cdot \mapsto \sigma^U\theta\}^{\emptyset}\sigma f(\cdot) && \text{if } FV(\sigma f(\cdot)) \cap U = \emptyset \\
\sigma^U(\theta + \eta) &= \sigma^U\theta + \sigma^U\eta \\
\sigma^U((\theta)') &= (\sigma^{\mathbb{V}}\theta)'
\end{aligned}
$$

$$
\begin{aligned}
\sigma^U(p(\theta)) &= (\sigma^U p)(\sigma^U \theta) && \text{if } FV(\sigma p(\cdot)) \cap U = \emptyset \\
\sigma^U(\phi \wedge \psi) &= \sigma^U\phi \wedge \sigma^U\psi \\
\sigma^U(\forall x\, \phi) &= \forall x\, \sigma^{U\cup\{x\}}\phi \\
\sigma^U(\langle\alpha\rangle\phi) &= \langle\sigma^U_V\alpha\rangle\sigma^V\phi
\end{aligned}
$$

$$
\begin{aligned}
\sigma^U_{U\cup BV(\sigma a)}(a) &= \sigma a \\
\sigma^U_{U\cup\{x\}}(x := \theta) &= x := \sigma^U\theta \\
\sigma^U_{U\cup\{x,x'\}}(x' = \theta \,\&\, q) &= (x' = \sigma^{U\cup\{x,x'\}}\theta \,\&\, \sigma^{U\cup\{x,x'\}}q) \\
\sigma^U_W(\alpha \cup \beta) &= \sigma^U_V\alpha \cup \sigma^U_W\beta \\
\sigma^U_W(\alpha;\beta) &= \sigma^U_V\alpha; \sigma^V_W\beta \\
\sigma^U_V(\alpha^*) &= (\sigma^V_V\alpha)^* && \text{where } \sigma^U_V\alpha \text{ defined} \\
\sigma^U_V(\alpha^d) &= (\sigma^U_V\alpha)^d
\end{aligned}
$$

input

output

$$\sigma^U(f(\theta)) = (\sigma^U f)(\sigma^U \theta)$$

$$\stackrel{\text{def}}{=} \{\cdot \mapsto \sigma^U \theta\}^{\emptyset} \sigma f(\cdot) \qquad \text{if } FV(\sigma f(\cdot)) \cap U = \emptyset$$

$$\sigma^U(\theta + \eta) = \sigma^U \theta + \sigma^U \eta$$

$$\sigma^U((\theta)') = (\sigma^{\mathbb{V}} \theta)'$$

$$\sigma^U(p(\theta)) = (\sigma^U p)(\sigma^U \theta) \qquad \text{if } FV(\sigma p(\cdot)) \cap U = \emptyset$$

$$\sigma^U(\phi \wedge \psi) = \sigma^U \phi \wedge \sigma^U \psi$$

$$\sigma^U(\forall x\, \phi) = \forall x\, \sigma^{U \cup \{x\}} \phi$$

Idea

Linear homomorphic pass postponing admissibility.
Recover with taboos at replacements.

$$\sigma^U_{U \cup \{x,x'\}}(x' = \theta \,\&\, q) = (x' = \sigma^{U \cup \{x,x'\}} \theta \,\&\, \sigma^{U \cup \{x,x'\}} q)$$

$$\sigma^U_{V \cup W}(\alpha \cup \beta) = \sigma^U_V \alpha \cup \sigma^U_W \beta$$

$$\sigma^U_W(\alpha; \beta) = \sigma^U_V \alpha; \sigma^V_W \beta$$

$$\sigma^U_V(\alpha^*) = (\sigma^V_V \alpha)^* \qquad \text{where } \sigma^U_V \alpha \text{ defined}$$

$$\sigma^U_V(\alpha^d) = (\sigma^U_V \alpha)^d$$

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

$$\text{US}\ \frac{\phi}{\sigma^{\emptyset}\phi}$$

provided $\sigma^{\emptyset}\phi$ is defined

If you bind a free variable, you go to logic jail!

Such a clash can only happen with taboos U arising while forming $\sigma^{\emptyset}\phi$

Soundness of Uniform Substitutions

"Syntactic uniform substitution = semantic replacement"

Lemma (Uniform substitution lemma)

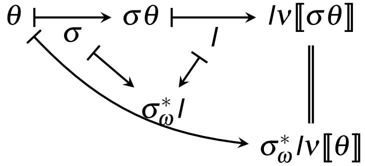
*Uniform substitution σ and adjoint $\sigma^*_\omega I$ to σ for I, ω have the same semantics for **all** ν such that $\nu = \omega$ except on U:*

$$I\nu[\![\sigma^U\theta]\!] = \sigma^*_\omega I\nu[\![\theta]\!]$$

$$\nu \in I[\![\sigma^U\phi]\!] \text{ iff } \nu \in \sigma^*_\omega I[\![\phi]\!]$$

$$\nu \in I[\![\sigma^U_V\alpha]\!](X) \text{ iff } \nu \in \sigma^*_\omega I[\![\alpha]\!](X)$$

Induction lexicographically on σ and $\phi + \alpha$ simultaneously,
with nested induction over closure ordinal, simultaneously for all ν, ω, U, X



Uniform Substitution of Rules

Theorem (Soundness)

$$\frac{\phi_1 \quad \dots \quad \phi_n}{\psi} \textit{ locally sound implies } \frac{\sigma^{\mathbb{V}}\phi_1 \quad \dots \quad \sigma^{\mathbb{V}}\phi_n}{\sigma^{\mathbb{V}}\psi} \textit{ locally sound}$$

Locally sound

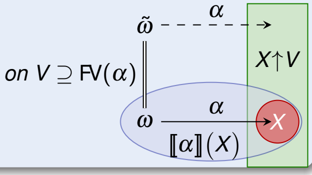
The conclusion is valid in any interpretation in which the premises are.

Lemma (Coincidence for formulas) (Only $FV(\phi)$ determine truth)

If $\omega = \tilde{\omega}$ *on* $FV(\phi)$ *then:* $\omega \in [\![\phi]\!]$ *iff* $\tilde{\omega} \in [\![\phi]\!]$

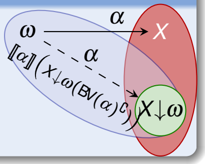
Lemma (Coincidence for games) (Only $FV(\alpha)$ determine victory)

If $\omega = \tilde{\omega}$ *on* $V \supseteq FV(\alpha)$ *then:*
$\omega \in [\![\alpha]\!](X \uparrow V)$ *iff* $\tilde{\omega} \in [\![\alpha]\!](X \uparrow V)$



Lemma (Bound effect) (Only $BV(\alpha)$ change value)

$\omega \in [\![\alpha]\!](X)$ *iff* $\omega \in [\![\alpha]\!](X \downarrow \omega(BV(\alpha)^\complement))$



Differential Game Logic: Axiomatization

Axiom = one formula

Infinite axiom schema

$[a]p(\bar{x}) \leftrightarrow \neg\langle a\rangle\neg p(\bar{x})$	$[\cdot]$	$[\alpha]\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$
$\langle x:=f\rangle p(x) \leftrightarrow p(f)$	$\langle:=\rangle$	$\langle x:=\theta\rangle\phi \leftrightarrow \phi_x^\theta$
$\langle x'=f\rangle p(x) \leftrightarrow \exists t{\geq}0\,\langle x:=x+ft\rangle p(x)$	$\langle'\rangle$	$\langle x'=\theta\rangle\phi \leftrightarrow \exists t{\geq}0\,\langle x:=y(t)\rangle\phi$
$\langle ?q\rangle p \leftrightarrow (q\wedge p)$	$\langle ?\rangle$	$\langle ?\psi\rangle\phi \leftrightarrow (\psi\wedge\phi)$
$\langle a\cup b\rangle p(\bar{x}) \leftrightarrow \langle a\rangle p(\bar{x}) \vee \langle b\rangle p(\bar{x})$	$\langle\cup\rangle$	$\langle\alpha\cup\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\phi\vee\langle\beta\rangle\phi$
$\langle a;b\rangle p(\bar{x}) \leftrightarrow \langle a\rangle\langle b\rangle p(\bar{x})$	$\langle ;\rangle$	$\langle\alpha;\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\langle\beta\rangle\phi$
$\langle a^*\rangle p(\bar{x}) \leftrightarrow p(\bar{x}) \vee \langle a\rangle\langle a^*\rangle p(\bar{x})$	$\langle^*\rangle$	$\langle\alpha^*\rangle\phi \leftrightarrow \phi\vee\langle\alpha\rangle\langle\alpha^*\rangle\phi$
$\langle a^d\rangle p(\bar{x}) \leftrightarrow \neg\langle a\rangle\neg p(\bar{x})$	$\langle^d\rangle$	$\langle\alpha^d\rangle\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$

IJCAR'18

TOCL'15

Axiom = one formula

Infinite axiom schema

$[a]\langle c\rangle\top \leftrightarrow \neg\langle a\rangle\neg\langle c\rangle\top$	$[\cdot]$	$[\alpha]\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$
$\langle x:=f\rangle\langle c\rangle\top \leftrightarrow \exists x\,(x=f \land \langle c\rangle\top)$	$\langle :=\rangle$	$\langle x:=\theta\rangle\phi \leftrightarrow \phi_x^\theta$
$\langle x'=f\rangle p(x) \leftrightarrow \exists t{\geq}0\,\langle x:=x+ft\rangle p(x)$	$\langle '\rangle$	$\langle x'=\theta\rangle\phi \leftrightarrow \exists t{\geq}0\,\langle x:=y(t)\rangle\phi$
$\langle ?q\rangle p \leftrightarrow (q \land p)$	$\langle ?\rangle$	$\langle ?\psi\rangle\phi \leftrightarrow (\psi \land \phi)$
$\langle a\cup b\rangle\langle c\rangle\top \leftrightarrow \langle a\rangle\langle c\rangle\top \lor \langle b\rangle\langle c\rangle\top$	$\langle\cup\rangle$	$\langle\alpha\cup\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\phi \lor \langle\beta\rangle\phi$
$\langle a;b\rangle\langle c\rangle\top \leftrightarrow \langle a\rangle\langle b\rangle\langle c\rangle\top$	$\langle ;\rangle$	$\langle\alpha;\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\langle\beta\rangle\phi$
$\langle a^*\rangle\langle c\rangle\top \leftrightarrow \langle c\rangle\top \lor \langle a\rangle\langle a^*\rangle\langle c\rangle\top$	$\langle {}^*\rangle$	$\langle\alpha^*\rangle\phi \leftrightarrow \phi \lor \langle\alpha\rangle\langle\alpha^*\rangle\phi$
$\langle a^d\rangle\langle c\rangle\top \leftrightarrow \neg\langle a\rangle\neg\langle c\rangle\top$	$\langle {}^d\rangle$	$\langle\alpha^d\rangle\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$

CADE'19

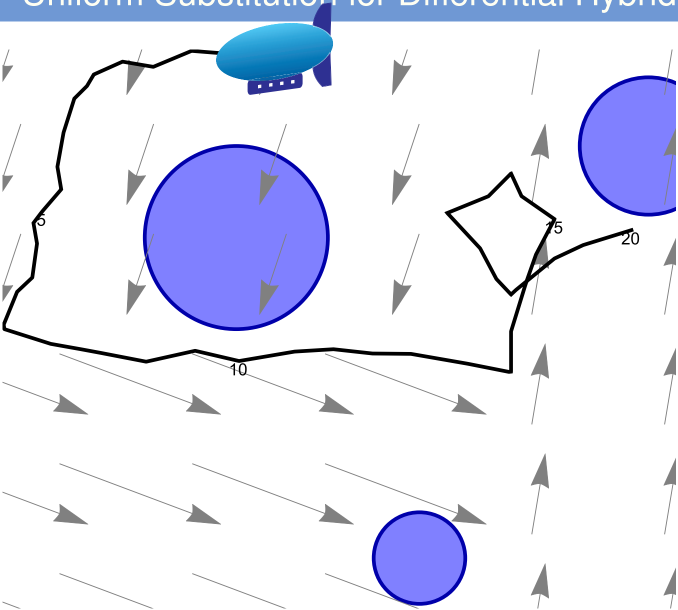
TOCL'15

Differential Game Logic: Axiomatization

$\langle c\rangle\top$ uniformly substitutes to $\langle ?\phi\rangle\top$ alias ϕ

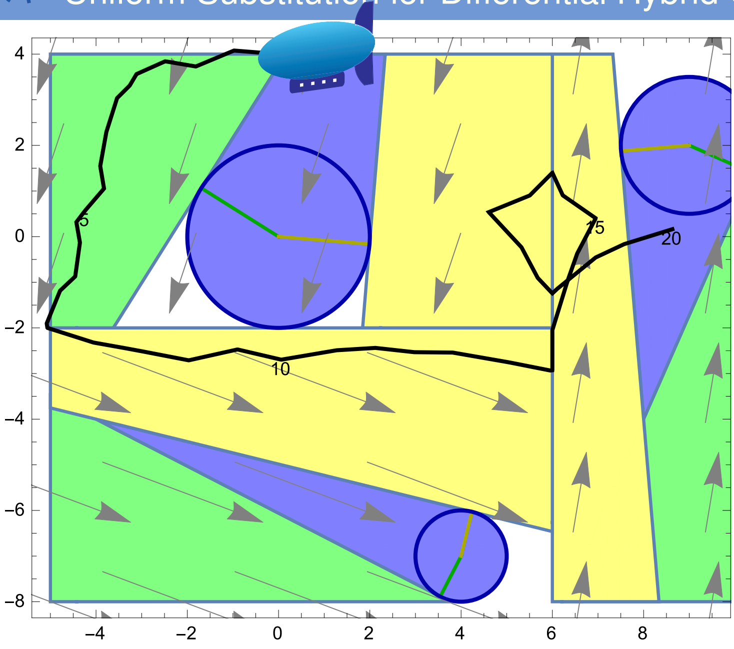
$[a]\langle c\rangle\top \leftrightarrow \neg\langle a\rangle\neg\langle c\rangle\top$	$[\cdot]$	$[\alpha]\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$
$\langle x:=f\rangle\langle c\rangle\top \leftrightarrow \exists x\,(x=f \wedge \langle c\rangle\top)$	$\langle:=\rangle$	$\langle x:=\theta\rangle\phi \leftrightarrow \phi_x^\theta$
$\langle x'=f\rangle p(x) \leftrightarrow \exists t{\geq}0\,\langle x:=x+ft\rangle p(x)$	$\langle'\rangle$	$\langle x'=\theta\rangle\phi \leftrightarrow \exists t{\geq}0\,\langle x:=y(t)\rangle\phi$
$\langle ?q\rangle p \leftrightarrow (q \wedge p)$	$\langle?\rangle$	$\langle ?\psi\rangle\phi \leftrightarrow (\psi \wedge \phi)$
$\langle a\cup b\rangle\langle c\rangle\top \leftrightarrow \langle a\rangle\langle c\rangle\top \vee \langle b\rangle\langle c\rangle\top$	$\langle\cup\rangle$	$\langle\alpha\cup\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\phi \vee \langle\beta\rangle\phi$
$\langle a;b\rangle\langle c\rangle\top \leftrightarrow \langle a\rangle\langle b\rangle\langle c\rangle\top$	$\langle;\rangle$	$\langle\alpha;\beta\rangle\phi \leftrightarrow \langle\alpha\rangle\langle\beta\rangle\phi$
$\langle a^*\rangle\langle c\rangle\top \leftrightarrow \langle c\rangle\top \vee \langle a\rangle\langle a^*\rangle\langle c\rangle\top$	$\langle^*\rangle$	$\langle\alpha^*\rangle\phi \leftrightarrow \phi \vee \langle\alpha\rangle\langle\alpha^*\rangle\phi$
$\langle a^d\rangle\langle c\rangle\top \leftrightarrow \neg\langle a\rangle\neg\langle c\rangle\top$	$\langle^d\rangle$	$\langle\alpha^d\rangle\phi \leftrightarrow \neg\langle\alpha\rangle\neg\phi$

CADE'19 TOCL'15



avoid obstacles
changing wind
local turbulence
$x' = f(x, y, z)$

TOCL'17



avoid obstacles
changing wind
local turbulence
$x' = f(x, y, z)$

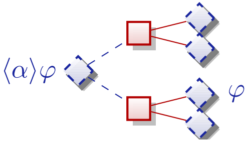
TOCL'17

Outline

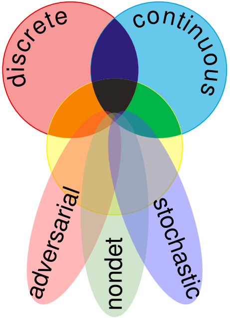
1. Motivation
   - Parsimonious Hybrid Game Proofs
   - Foundation for Verification
2. Differential Game Logic
   - Syntax
   - Example: Push-around Cart
   - Denotational Semantics
3. Uniform Substitution
   - Application
   - Uniform Substitution Lemma
   - Uniform Substitution of Rules
   - Static Semantics
   - Axioms
   - Differential Hybrid Games
4. Summary

differential game logic

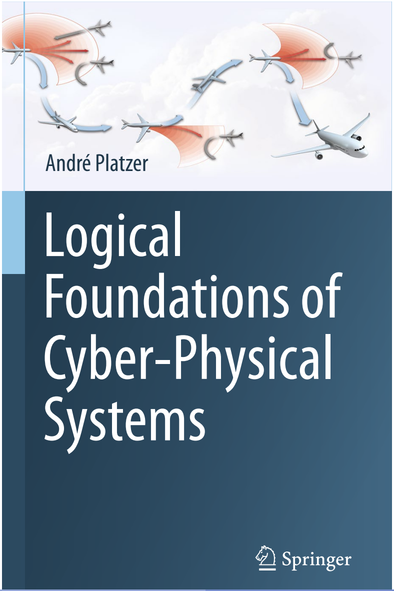
$\mathsf{dGL} = \mathsf{GL} + \mathsf{HG} = \mathsf{dL} + {}^d$



- Faster sound uniform substitution
- Replace all at once, check all at once
- Modular: Logic $\|$ Prover
- Isabelle/HOL formalization 3,500 lines
- Sound & rel. complete axiomatization
- Sound for differential hybrid games
- Future: Benefit from USubst elsewhere



A. Platzer. *Logical Foundations of Cyber-Physical Systems*. Springer 2018



Foundation for Verification

Foundation for	FOL	Functional Language	Imperative Language
↑	Formula	Functional program	Imperative program/game
↑	Predicate calculus	Function calculus	Program calculus
↑	Subst + rename	α, β, η-conversion	USubst + rename

Functional

α-conversion	for bound variables
β-reduction	capture-avoiding subst.
η-conversion	versus free variables

Imperative

Uniform substitution replaces predicate/function/program sym. mindful of free/bound variables

Substitution is fundamental but subtle. Henkin wants it banished!

Now: Make USubst even more subtle, but faster, and still sound.

Beware: Imperative free and bound variables may overlap!

André Platzer.
Uniform substitution at one fell swoop.
In Pascal Fontaine, editor, *CADE*, volume 11716 of *LNCS*, pages 425–441. Springer, 2019.
doi:10.1007/978-3-030-29436-6_25.

André Platzer.
Uniform substitution for differential game logic.
In Didier Galmiche, Stephan Schulz, and Roberto Sebastiani, editors, *IJCAR*, volume 10900 of *LNCS*, pages 211–227. Springer, 2018.
doi:10.1007/978-3-319-94205-6_15.

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

André Platzer.
Differential hybrid games.
ACM Trans. Comput. Log., 18(3):19:1–19:44, 2017.
doi:10.1145/3091123.

André Platzer.
Logical Foundations of Cyber-Physical Systems.
Springer, Cham, 2018.
doi:10.1007/978-3-319-63588-0.

André Platzer.
A complete uniform substitution calculus for differential dynamic logic.
J. Autom. Reas., 59(2):219–265, 2017.
doi:10.1007/s10817-016-9385-1.

Outline

5 Appendix
- ODE Schema
- Static Semantics
- Operational Semantics
- Completeness

Axiom Schemata Need Side Conditions: Solving ODEs

$\langle ' \rangle \quad \langle x' = \theta \rangle \phi \leftrightarrow \exists t \geq 0 \, \langle x := y(t) \rangle \phi$

Axiom schema with side conditions:

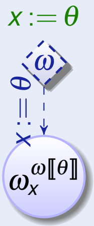
1. Occurs check: t fresh
2. Solution check: $y(\cdot)$ solves the ODE $y'(t) = \theta$ with $x(\cdot)$ plugged in for x in term θ
3. Initial value check: $y(\cdot)$ solves the symbolic IVP $y(0) = x$
4. $x(\cdot)$ covers all solutions parametrically
5. x' cannot occur free in ϕ

Quite nontrivial soundness-critical algorithms . . .

Static Semantics

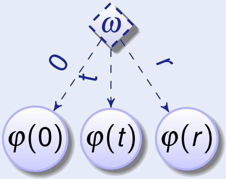
$$\mathsf{FV}(\theta) = \{x \in \mathbb{V} : \exists\, \omega, \tilde{\omega} \text{ such that } \omega = \tilde{\omega} \text{ on } \{x\}^\complement \text{ and } \omega[\![\theta]\!] \neq \tilde{\omega}[\![\theta]\!]\}$$
$$\mathsf{FV}(\phi) = \{x \in \mathbb{V} : \exists\, \omega, \tilde{\omega} \text{ such that } \omega = \tilde{\omega} \text{ on } \{x\}^\complement \text{ and } \omega \in [\![\phi]\!] \not\ni \tilde{\omega}\}$$
$$\mathsf{FV}(\alpha) = \{x \in \mathbb{V} : \exists\, \omega, \tilde{\omega}, X \text{ with } \omega = \tilde{\omega} \text{ on } \{x\}^\complement,\ \omega \in [\![\alpha]\!](X{\uparrow}\{x\}^\complement) \not\ni \tilde{\omega}\}$$
$$\mathsf{BV}(\alpha) = \{x \in \mathbb{V} : \exists\, \omega, X \text{ such that } [\![\alpha]\!](X) \ni \omega \notin [\![\alpha]\!](X{\downarrow}\omega(\{x\}))\}$$

Definition (Hybrid game α: operational semantics)



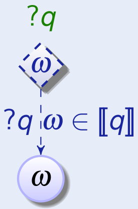
Definition (Hybrid game α: operational semantics)

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



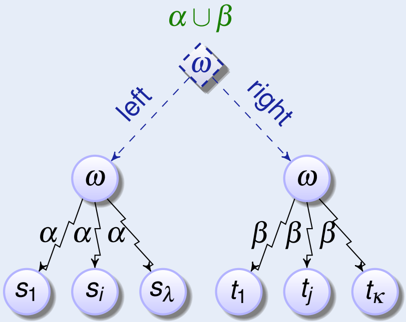
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)

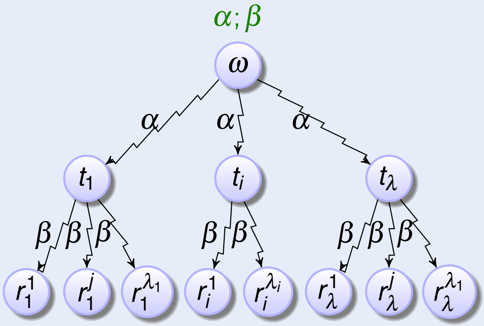


Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)

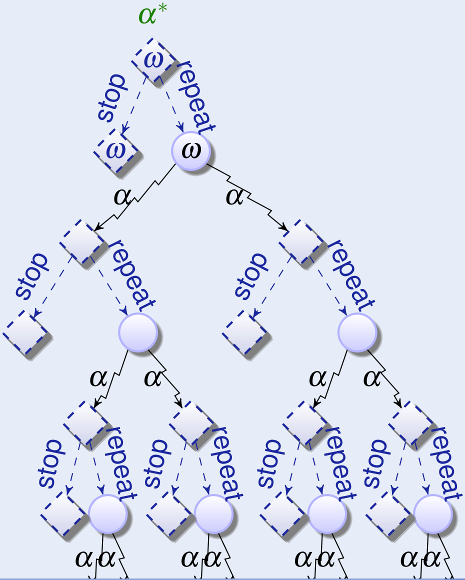


Definition (Hybrid game α: operational semantics)

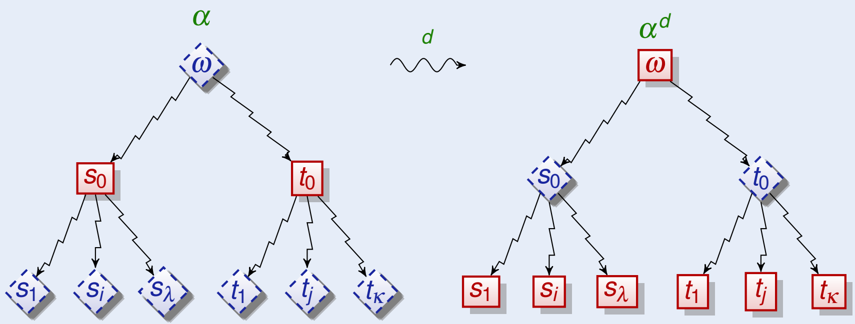


Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



Definition (Hybrid game α: operational semantics)



Theorem (Completeness)

dGL *calculus is a sound & complete axiomatization relative to any (differentially) expressive*[1] *logic L.*

$$\vDash \varphi \quad \textit{iff} \quad Taut_L \vdash \varphi$$

[1] $\forall \varphi \in \mathrm{dGL}\ \exists \varphi^\flat \in L \quad \vDash \varphi \leftrightarrow \varphi^\flat$
$\langle x' = \theta \rangle G \leftrightarrow (\langle x' = \theta \rangle G)^\flat$ provable for $G \in L$

TOCL'15