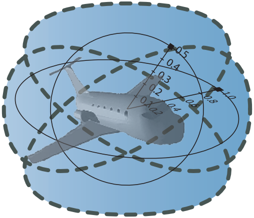
16: Hybrid Systems & Games

15-424: Foundations of Cyber-Physical Systems

André Platzer

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



Outline

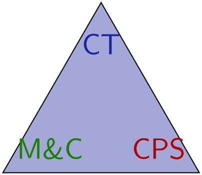
1. Learning Objectives
2. Motivation
3. Hybrid Games
    - Choices & Nondeterminism
    - Control & Dual Control
    - Hybrid Games
    - Differential Game Logic
    - Demon's Controls
    - Operational Game Semantics
    - Filibusters & Finitude
4. Example: Robot Factory
5. Summary

Outline

Learning Objectives

Hybrid Systems & Games

fundamental principles of computational thinking
logical extensions
PL modularity principles
compositional extensions
differential game logic
best-worst-case analysis
models of alternating computation



adversarial dynamics
conflicting actions
multi-agent systems
angelic/demonic choice

multi-agent state change
CPS semantics
reflections on choices

Outline

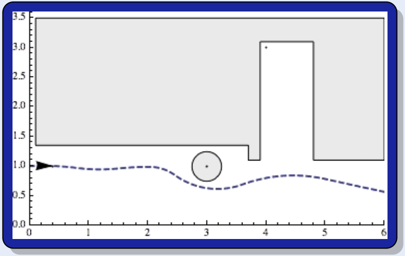
1. Learning Objectives
2. Motivation
3. Hybrid Games
   - Choices & Nondeterminism
   - Control & Dual Control
   - Hybrid Games
   - Differential Game Logic
   - Demon's Controls
   - Operational Game Semantics
   - Filibusters & Finitude
4. Example: Robot Factory
5. Summary

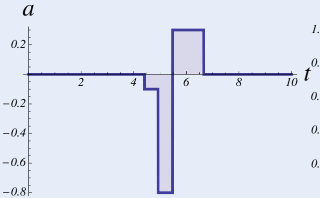
CPS Analysis: Robot Control

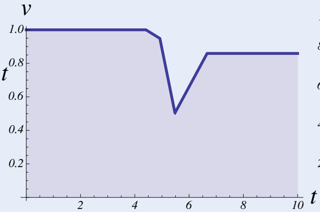
Challenge (Hybrid Systems)

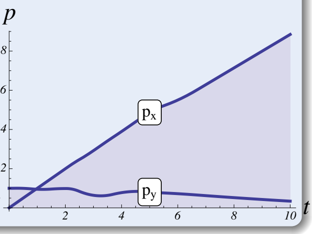
Fixed rule describing state evolution with both

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







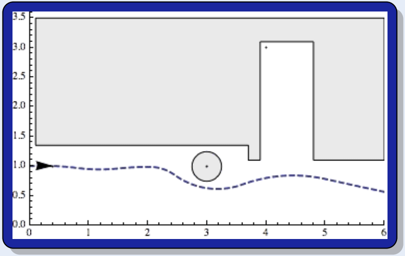


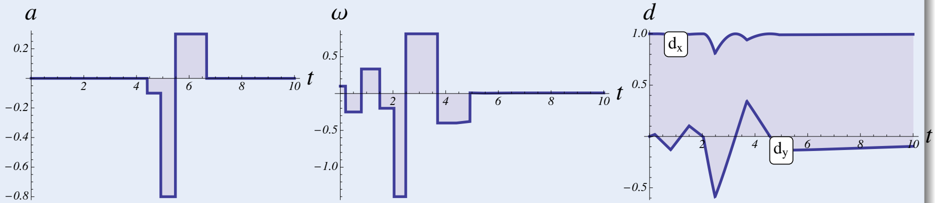
CPS Analysis: Robot Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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



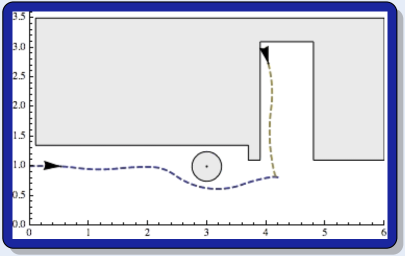


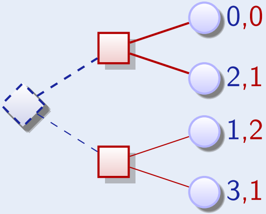
CPS Analysis: Robot Control

Challenge (Games)

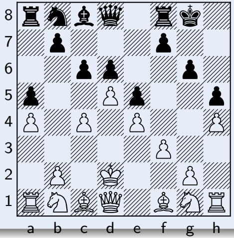
Game rules describing play evolution with both

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





◇\□	Tr	Pl
Trash	1,2	0,0
Plant	0,0	2,1

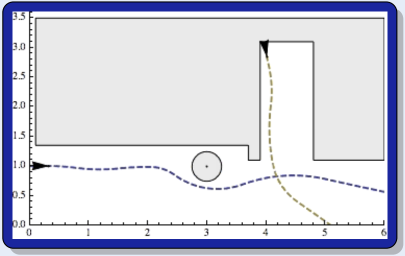


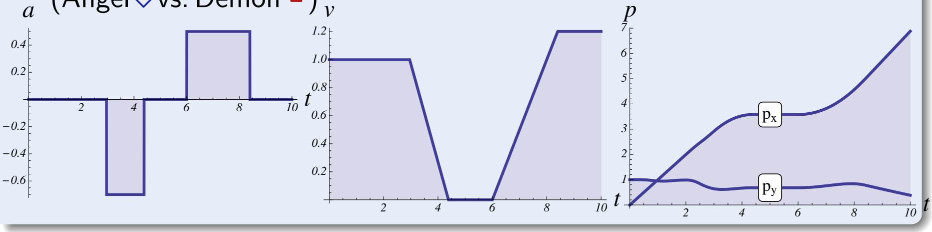
CPS Analysis: Robot Control

Challenge (Hybrid Games)

Game rules describing play evolution with

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



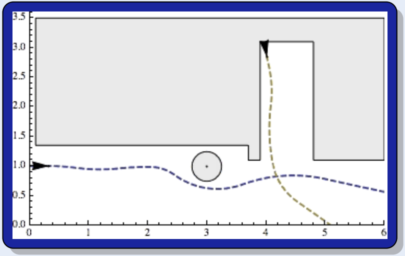


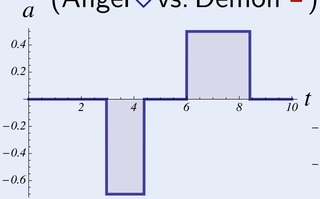
CPS Analysis: Robot Control

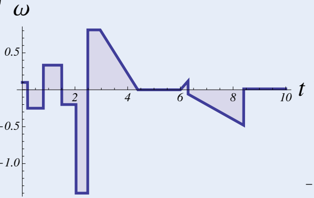
Challenge (Hybrid Games)

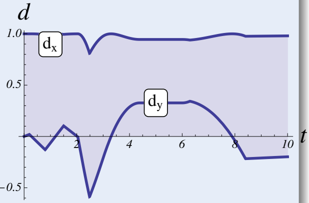
Game rules describing play evolution with

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







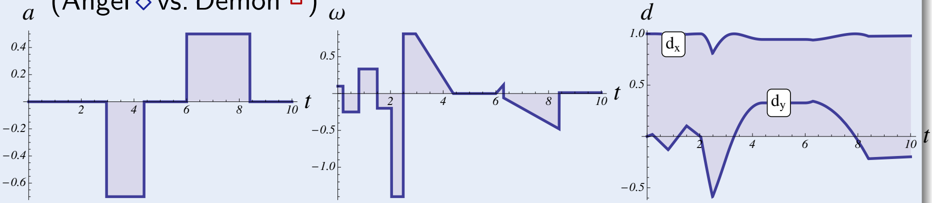


CPS Analysis: RoboCup Soccer

Challenge (Hybrid Games)

Game rules describing play evolution with

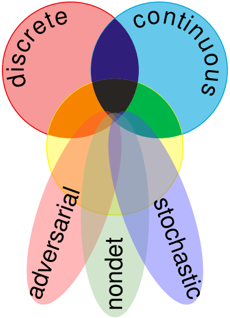
- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)
- Adversarial dynamics (Angel ◇ vs. Demon □)



CPSs are Multi-Dynamical Systems

CPS Dynamics

CPS are characterized by multiple facets of dynamical systems.



CPS Compositions

CPS combine multiple simple dynamical effects.

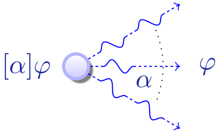
Tame Parts

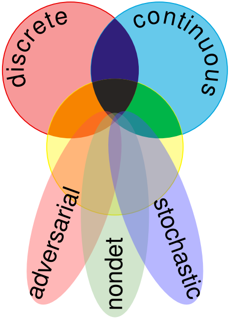
Exploiting compositionality tames CPS complexity.

Dynamic Logics for Dynamical Systems

differential dynamic logic

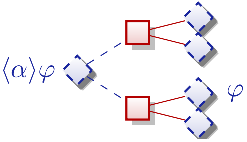
$d\mathcal{L} = \mathsf{DL} + \mathsf{HP}$





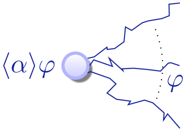
differential game logic

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



stochastic differential DL

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



quantified differential DL

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

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

LICS'12,CADE'15,TOCL'15

Outline

1. Learning Objectives
2. Motivation
3. Hybrid Games
    - Choices & Nondeterminism
    - Control & Dual Control
    - Hybrid Games
    - Differential Game Logic
    - Demon's Controls
    - Operational Game Semantics
    - Filibusters & Finitude
4. Example: Robot Factory
5. Summary

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

Definition (Hybrid program α)

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

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

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

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

Discrete Assign

Test Condition

Differential Equation

Nondet. Choice

Seq. Compose

Nondet. Repeat

Definition (Hybrid program α)

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

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

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

All Reals

Some Reals

All Runs

Some Runs

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

Nondet. Choice

Definition (Hybrid program α)

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

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

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

Nondeterminism during HP runs

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

Differential Equation

Nondet. Choice

Nondet. Repeat

Definition (Hybrid program α)

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

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

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

Nondeterminism during HP runs

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

Differential Equation

Nondet. Choice

Nondet. Repeat

Definition (Hybrid program α)

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

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 Choices

Some Choice

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

All choices resolved in one way

Differential Equation

Nondet. Choice

Nondet. Repeat

Definition (Hybrid program α)

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

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

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

Modality decides the mode: help/hurt

All Choices

Some Choice

Game Operators

◇ Angel Ops

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

Let Angel be a player

Game Operators

◇ Angel Ops

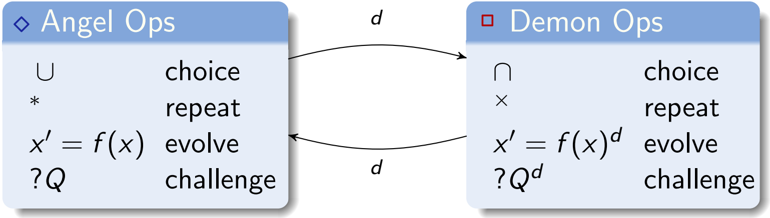
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

□ Demon Ops

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

Let Demon be another player

Game Operators



◇ Angel Ops

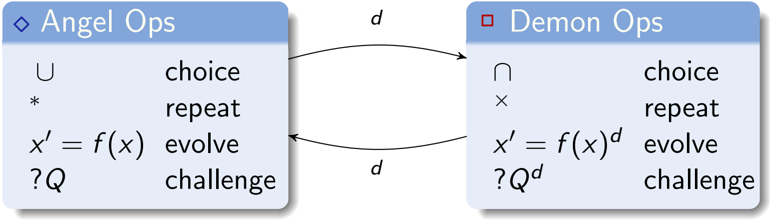
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

□ Demon Ops

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

Duality operator ^{d} passes control between players

Game Operators



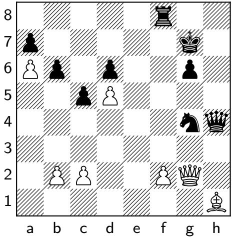
◇ Angel Ops

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

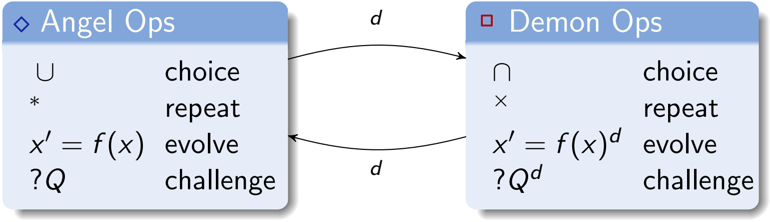
☐ Demon Ops

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

Duality operator ^{d} passes control between players



Game Operators



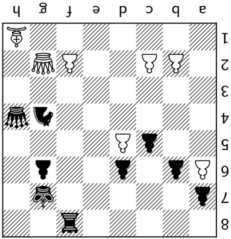
◇ Angel Ops

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

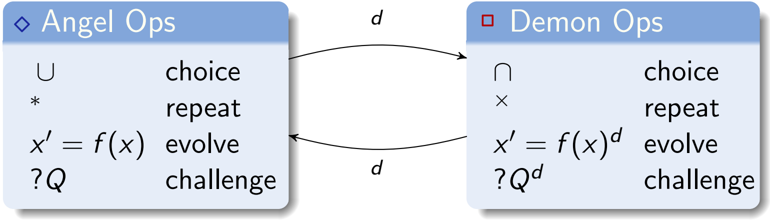
◻ Demon Ops

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

Duality operator ^{d} passes control between players



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

Differential Game Logic: Syntax

Definition (Hybrid game α)

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

TOCL'15

Differential Game Logic: Syntax

Definition (Hybrid game α)

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

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

$$p(e_1, \dots, e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle \alpha \rangle P \mid [\alpha] P$$

TOCL'15

Differential Game Logic: Syntax

Discrete Assign

Test Game

Differential Equation

Choice Game

Seq. Game

Repeat Game

Definition (Hybrid game α)

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

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

$$p(e_1,\dots,e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \land Q \mid \forall x\, P \mid \exists x\, P \mid \langle\alpha\rangle P \mid [\alpha]P$$

All Reals

Some Reals

TOCL'15

Differential Game Logic: Syntax

Discrete Assign

Test Game

Differential Equation

Choice Game

Seq. Game

Repeat Game

Dual Game

Definition (Hybrid game α)

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

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

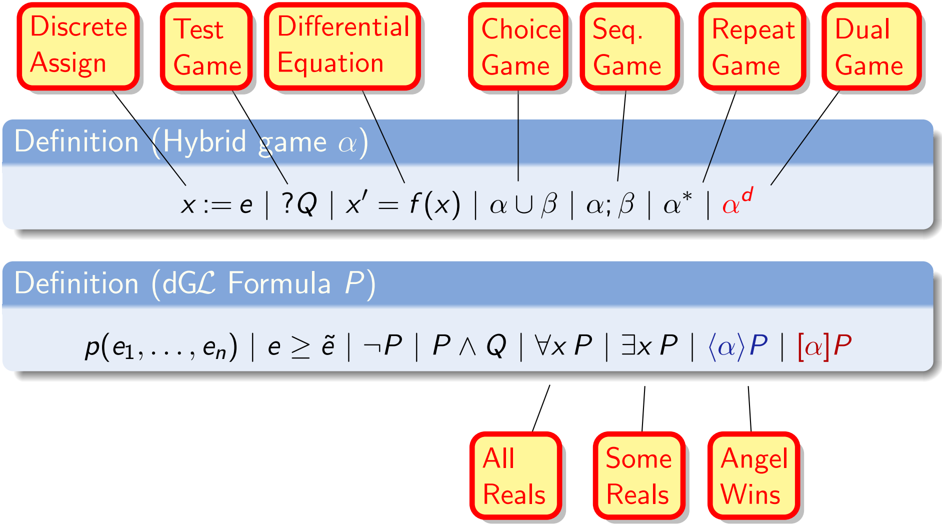
$$p(e_1,\dots,e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \land Q \mid \forall x\, P \mid \exists x\, P \mid \langle\alpha\rangle P \mid [\alpha]P$$

All Reals

Some Reals

TOCL'15

Differential Game Logic: Syntax



Definition (Hybrid game α)

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

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

$$p(e_1,\ldots,e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle\alpha\rangle P \mid [\alpha]P$$

TOCL'15

Differential Game Logic: Syntax

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

Definition (Hybrid game α)

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

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

$$p(e_1, \dots, e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle \alpha \rangle P \mid [\alpha] P$$

All Reals | Some Reals | Angel Wins | Demon Wins

TOCL'15

Simple Examples

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

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

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

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

Simple Examples

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

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

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

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

Simple Examples

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

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

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

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

Simple Examples

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

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

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

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

Simple Examples

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

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

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

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

Simple Examples

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

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

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

$\vDash \langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \leq x < 1)$
since stuck at $x = 0$ which wins

Simple Examples

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

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

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

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

since stuck at $x = 0$ which wins

Simple Examples

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

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

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

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

since stuck at $x = 0$ which wins

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

Simple Examples

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

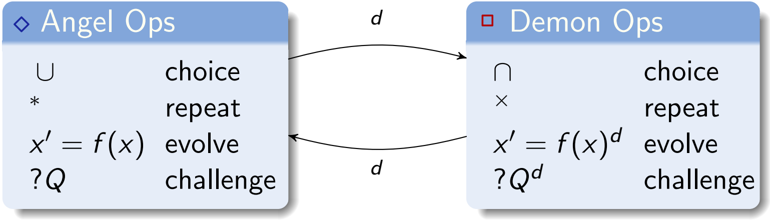
$\vDash \langle (x := x+1; (x' = x^2)^d \cup x := x-1)^* \rangle\, (0 \leq x < 1)$

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

$\vDash \langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 \leq x < 1)$
since stuck at $x = 0$ which wins

$\nvDash \langle (x := x+1; (x' = x^2)^d \cup (x := x-1 \cap x := x-2))^* \rangle (0 < x \leq 1)$
since stuck at $x = 0$ which needs another iteration and then loses

Definable Game Operators



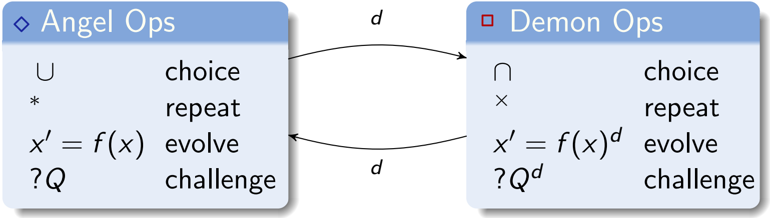
◇ Angel Ops

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

□ Demon Ops

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

Definable Game Operators



◇ Angel Ops

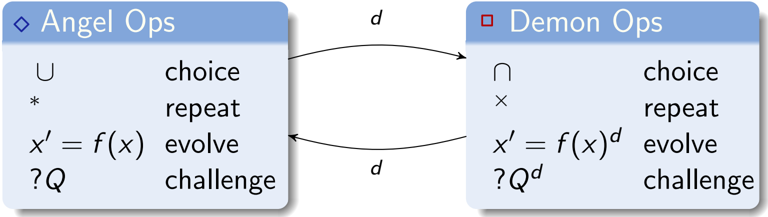
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

☐ Demon Ops

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

$$\begin{aligned}
\mathtt{if}(Q)\,\alpha\,\mathtt{else}\,\beta &\equiv \\
\mathtt{while}(Q)\,\alpha &\equiv \\
\alpha \cap \beta &\equiv \\
\alpha^{\times} &\equiv \\
(x' = f(x) \,\&\, Q)^d & \quad x' = f(x) \,\&\, Q \\
(x := e)^d & \quad x := e \\
?Q^d & \quad ?Q
\end{aligned}$$

Definable Game Operators



◇ Angel Ops

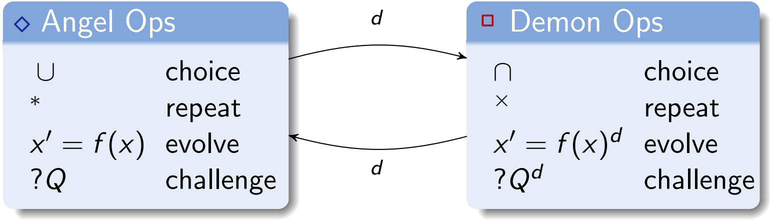
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

◻ Demon Ops

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

$$\begin{aligned}
\mathtt{if}(Q)\,\alpha\,\mathtt{else}\,\beta &\equiv (?Q;\alpha) \cup (?\neg Q;\beta)\\
\mathtt{while}(Q)\,\alpha &\equiv\\
\alpha \cap \beta &\equiv\\
\alpha^{\times} &\equiv\\
(x' = f(x)\,\&\,Q)^d &\qquad x' = f(x)\,\&\,Q\\
(x := e)^d &\qquad x := e\\
?Q^d &\qquad ?Q
\end{aligned}$$

Definable Game Operators



◇ Angel Ops

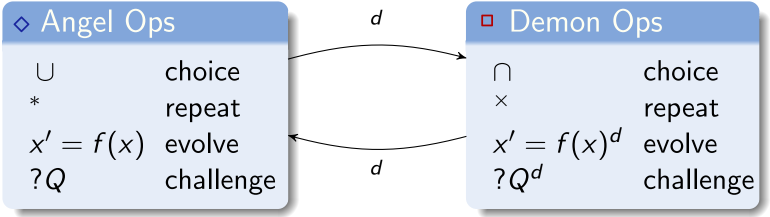
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

◻ Demon Ops

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

$$
\begin{aligned}
\texttt{if}(Q)\,\alpha\,\texttt{else}\,\beta &\equiv (?Q;\alpha) \cup (?\neg Q;\beta)\\
\texttt{while}(Q)\,\alpha &\equiv (?Q;\alpha)^*;?\neg Q\\
\alpha \cap \beta &\equiv\\
\alpha^\times &\equiv\\
(x' = f(x) \,\&\, Q)^d &\quad x' = f(x) \,\&\, Q\\
(x := e)^d &\quad x := e\\
?Q^d &\quad ?Q
\end{aligned}
$$

Definable Game Operators



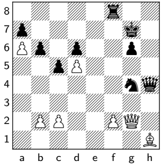
◇ Angel Ops

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

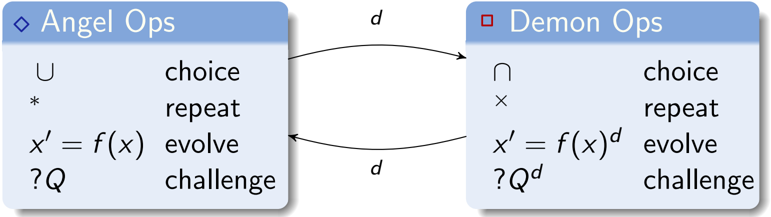
□ Demon Ops

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

$$\begin{aligned}
\mathtt{if}(Q)\,\alpha\,\mathtt{else}\,\beta &\equiv (?Q;\alpha) \cup (?\neg Q;\beta) \\
\mathtt{while}(Q)\,\alpha &\equiv (?Q;\alpha)^*;?\neg Q \\
\alpha \cap \beta &\equiv \\
\alpha^\times &\equiv \\
(x' = f(x)\,\&\,Q)^d & \quad x' = f(x)\,\&\,Q \\
(x := e)^d & \quad x := e \\
?Q^d & \quad ?Q
\end{aligned}$$



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

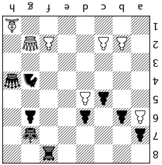
$$\texttt{if}(Q)\,\alpha\,\texttt{else}\,\beta \equiv (?Q;\alpha) \cup (?\neg Q;\beta)$$

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

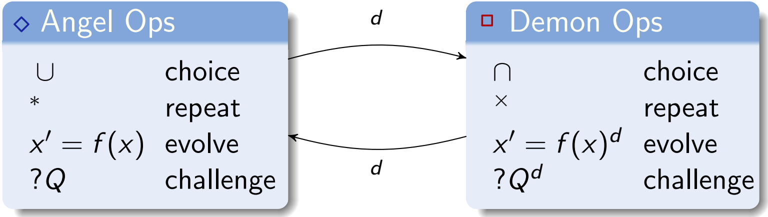
$$\alpha \cap \beta \equiv (\alpha^d \cup \beta^d)^d$$

$$\alpha^\times \equiv$$

$$\begin{array}{rl} (x' = f(x)\,\&\,Q)^d & x' = f(x)\,\&\,Q \\ (x := e)^d & x := e \\ ?Q^d & ?Q \end{array}$$



Definable Game Operators



◇ Angel Ops

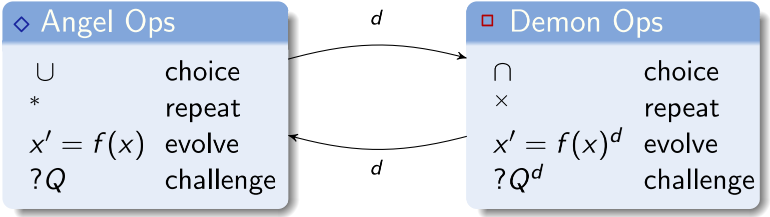
$\cup$	choice
$*$	repeat
$x' = f(x)$	evolve
$?Q$	challenge

□ Demon Ops

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

$$\begin{aligned}
\mathtt{if}(Q)\,\alpha\,\mathtt{else}\,\beta &\equiv (?Q;\alpha) \cup (?\neg Q;\beta)\\
\mathtt{while}(Q)\,\alpha &\equiv (?Q;\alpha)^*;?\neg Q\\
\alpha \cap \beta &\equiv (\alpha^d \cup \beta^d)^d\\
\alpha^\times &\equiv ((\alpha^d)^*)^d\\
(x' = f(x)\,\&\,Q)^d & \quad x' = f(x)\,\&\,Q\\
(x := e)^d & \quad x := e\\
?Q^d & \quad ?Q
\end{aligned}$$

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

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

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

$$\alpha \cap \beta \equiv (\alpha^d \cup \beta^d)^d$$

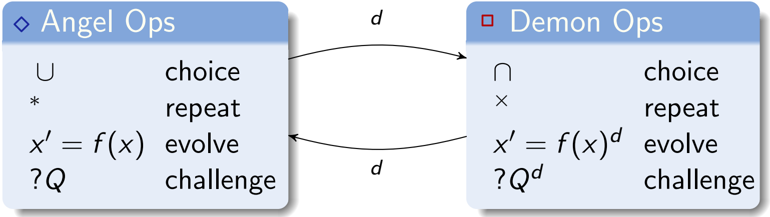
$$\alpha^\times \equiv ((\alpha^d)^*)^d$$

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

$$(x := e)^d \qquad x := e$$

$$?Q^d \qquad ?Q$$

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

$$\mathtt{if}(Q)\,\alpha\,\mathtt{else}\,\beta \equiv (?Q;\alpha) \cup (?\neg Q;\beta)$$

$$\mathtt{while}(Q)\,\alpha \equiv (?Q;\alpha)^*;?\neg Q$$

$$\alpha \cap \beta \equiv (\alpha^d \cup \beta^d)^d$$

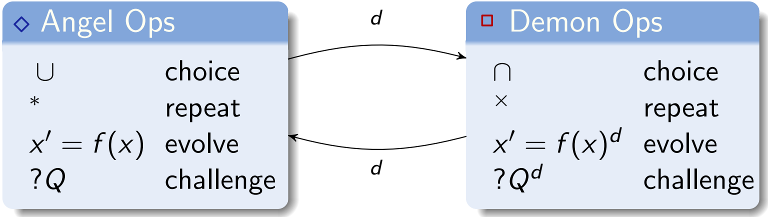
$$\alpha^\times \equiv ((\alpha^d)^*)^d$$

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

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

$$?Q^d \qquad ?Q$$

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

$$\begin{aligned}
\texttt{if}(Q)\,\alpha\,\texttt{else}\,\beta &\equiv (?Q;\alpha)\cup(?\neg Q;\beta)\\
\texttt{while}(Q)\,\alpha &\equiv (?Q;\alpha)^*;?\neg Q\\
\alpha\cap\beta &\equiv (\alpha^d\cup\beta^d)^d\\
\alpha^\times &\equiv ((\alpha^d)^*)^d\\
(x'=f(x)\,\&\,Q)^d &\not\equiv x'=f(x)\,\&\,Q\\
(x:=e)^d &\equiv x:=e\\
?Q^d &\not\equiv ?Q
\end{aligned}$$

Simple Examples: EVE and WALL·E

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

EVE at e plays Angel's part controlling g

WALL·E at w plays Demon's part controlling u

Simple Examples: EVE and WALL·E and the World

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

EVE at e plays Angel's part controlling g

WALL·E at w plays Demon's part controlling u

EVE assigned environment's time to WALL·E

Simple Examples: WALL·E and EVE

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

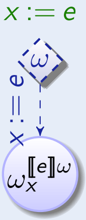
WALL·E at w plays Demon's part controlling u

EVE at e plays Angel's part controlling g

WALL·E assigned environment's time to EVE

Differential Game Logic: Operational Semantics

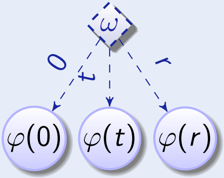
Definition (Hybrid game α: operational semantics)



Differential Game Logic: Operational Semantics

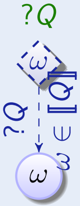
Definition (Hybrid game α: operational semantics)

$x' = f(x) \,\&\, Q$



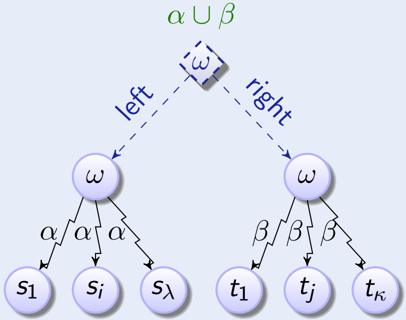
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



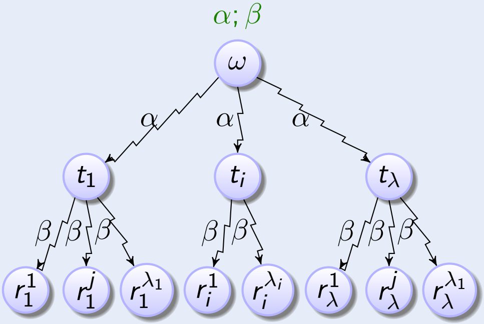
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



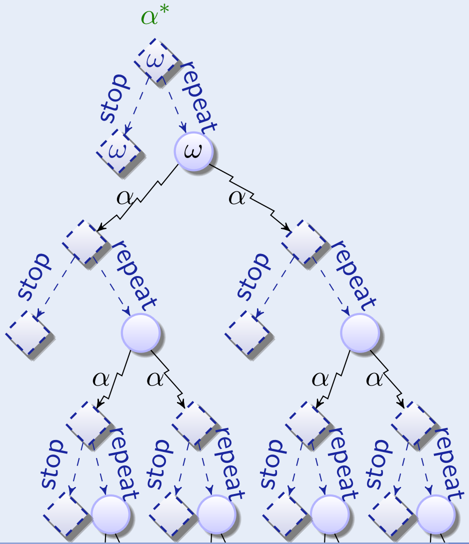
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



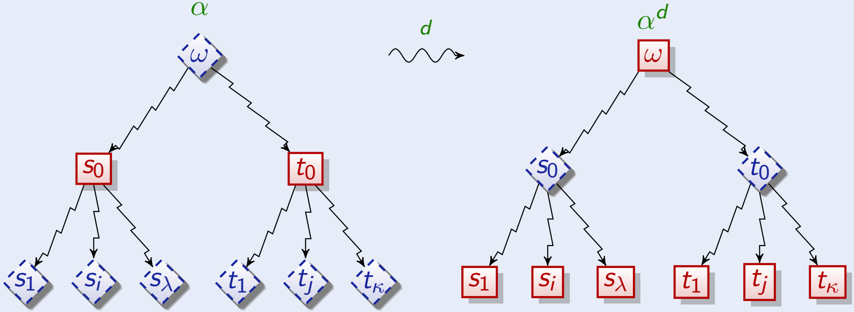
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



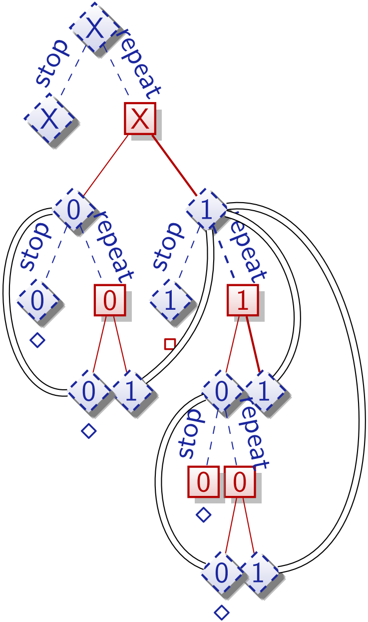
Differential Game Logic: Operational Semantics

Definition (Hybrid game α: operational semantics)



Filibusters

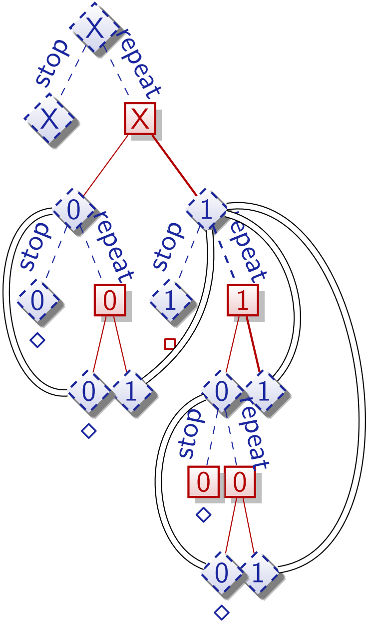
$$\langle(x := 0 \cap x := 1)^*\rangle x = 0$$



Filibusters & The Significance of Finitude

$\langle(x := 0 \cap x := 1)^*\rangle x = 0$

$\overset{\text{wfd}}{\rightsquigarrow}$ false unless $x = 0$



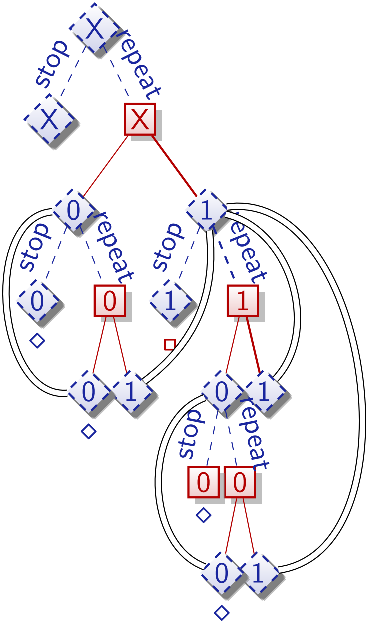
Filibusters & The Significance of Finitude

$$\langle (x' = 1^d; x := 0)^* \rangle x = 0$$

$$\langle (x := 0; x' = 1^d)^* \rangle x = 0$$

$$\langle (x := 0 \cap x := 1)^* \rangle x = 0$$

$\overset{\text{wfd}}{\rightsquigarrow}$ false unless $x = 0$



Filibusters & The Significance of Finitude

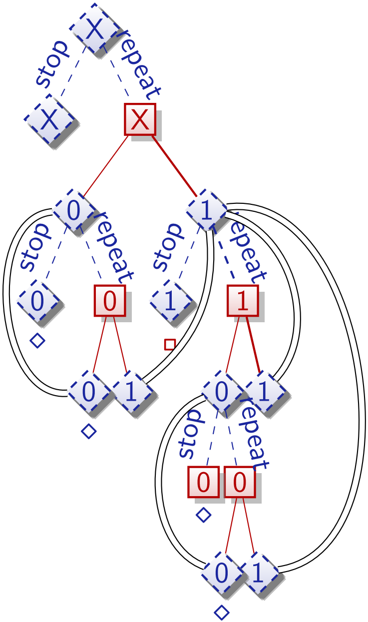
$\overset{<\infty}{\rightsquigarrow}$ true

$$\langle (x' = 1^d ; x := 0)^* \rangle x = 0$$

$$\langle (x := 0 ; x' = 1^d)^* \rangle x = 0$$

$$\langle (x := 0 \cap x := 1)^* \rangle x = 0$$

$\overset{\text{wfd}}{\rightsquigarrow}$ false unless $x = 0$



Filibusters & The Significance of Finitude

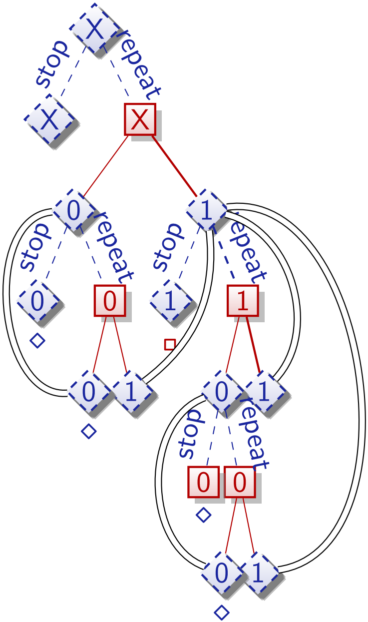
$\overset{<\infty}{\rightsquigarrow}$ true

$$\langle (x' = 1^d ; x := 0)^* \rangle x = 0$$

$$\langle (x := 0 ; x' = 1^d)^* \rangle x = 0$$

$$\langle (x := 0 \cap x := 1)^* \rangle x = 0$$

$\overset{\text{wfd}}{\rightsquigarrow}$ false unless $x = 0$

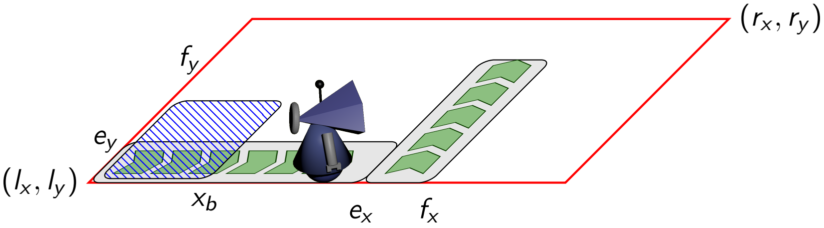


Well-defined games
can't be postponed forever

Outline

1. Learning Objectives
2. Motivation
3. Hybrid Games
   - Choices & Nondeterminism
   - Control & Dual Control
   - Hybrid Games
   - Differential Game Logic
   - Demon's Controls
   - Operational Game Semantics
   - Filibusters & Finitude
4. Example: Robot Factory
5. Summary

Example: Robot Factory Decentralized Automation



Model

- (x, y) robot coordinates
- (v_x, v_y) velocities
- conveyor belts may instantaneously increase robot's velocity by (c_x, c_y)

Primary objectives of the robot

- Leave ▨ within time ε
- Never leave outer □

Challenges

- Distributed, physical environment
- Possibly conflicting secondary objectives

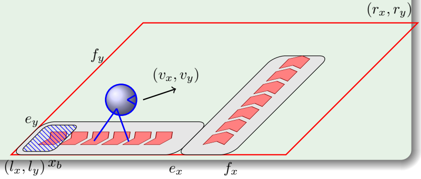
Robot Factory Automation (*RF*)

Example (Robot-Demon vs. Angel-Factory Environment)

$$\Big(\big(?true \cup (?(x < e_x \land y < e_y \land \text{eff}_1 = 1);\ v_x := v_x + c_x;\ \text{eff}_1 := 0) \qquad // \text{ belt}$$

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

$$\Big)^*$$



CADE'12

Robot Factory Automation (RF)

Example (Robot-Demon vs. Angel-Factory Environment)

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

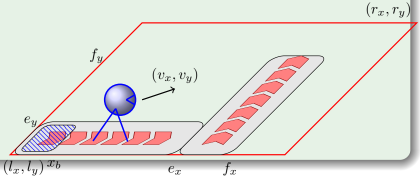
$$\cup\,(?(e_x \le x \land y \le f_y \land \mathrm{eff}_2 = 1);\ v_y := v_y + c_y;\ \mathrm{eff}_2 := 0)\,\big)\,;$$

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

$$a_y := *;\ ?(-A \le a_y \le A); \qquad // \text{ “independent” robot acceleration}$$

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

$$\Big)^*$$



CADE'12

Robot Factory Automation (*RF*)

Example (Robot-Demon vs. Angel-Factory Environment)

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

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

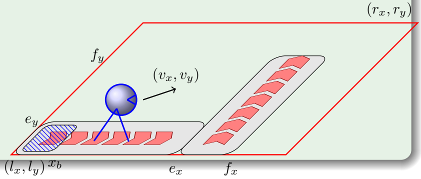
$$(\,a_x := *;\ ?(-A \le a_x \le A);$$

$$a_y := *;\ ?(-A \le a_y \le A); \qquad \text{// “independent” robot acceleration}$$

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

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

$$\Big)^*$$



CADE'12

Robot Factory Automation (*RF*)

Example (Robot-Demon vs. Angel-Factory Environment)

$$\Big(\big(?true \cup (?(x < e_x \wedge y < e_y \wedge \mathrm{eff}_1 = 1);\ v_x := v_x + c_x;\ \mathrm{eff}_1 := 0) \qquad \text{// belt}$$

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

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

$$a_y := *;\ ?(-A \leq a_y \leq A); \qquad \text{// ``independent'' robot acceleration}$$

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

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

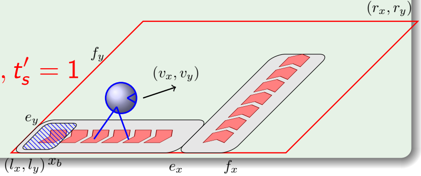
$$\cap\,(?(a_x v_x \leq 0 \wedge a_y v_y \leq 0)^d; \qquad \text{// brake}$$

$$\text{if } v_x = 0 \text{ then } a_x := 0 \text{ fi}; \qquad \text{// per direction: no time lock}$$

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

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

$$\&\ t_s \leq \varepsilon \wedge a_x v_x \leq 0 \wedge a_y v_y \leq 0))\big)\Big)^*$$



CADE'12

Robot Factory Automation (*RF*)

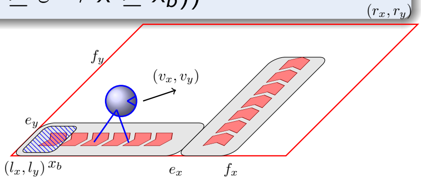
Proposition (Robot stays in □)

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

Proposition (Stays in □ and leaves ▨ on time)

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

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

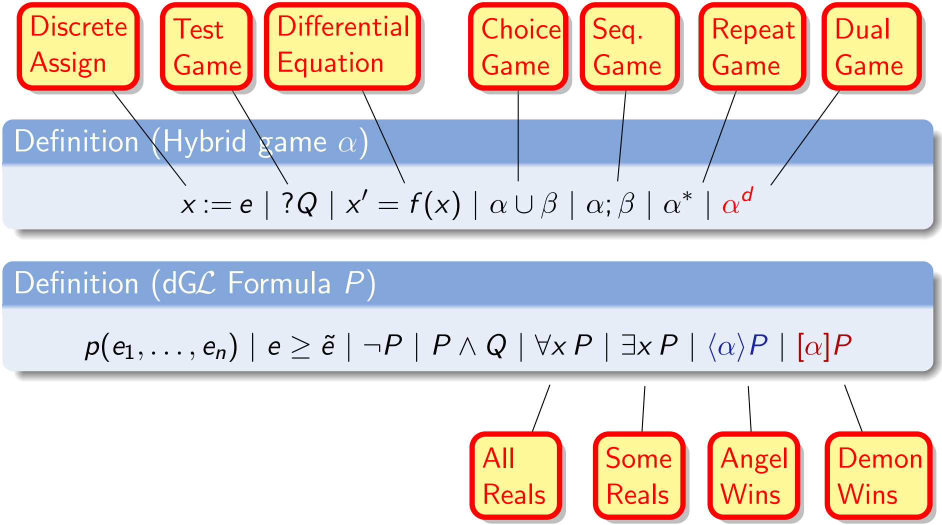


CADE'12

Outline

1. Learning Objectives
2. Motivation
3. Hybrid Games
    - Choices & Nondeterminism
    - Control & Dual Control
    - Hybrid Games
    - Differential Game Logic
    - Demon's Controls
    - Operational Game Semantics
    - Filibusters & Finitude
4. Example: Robot Factory
5. **Summary**

Differential Game Logic: Syntax



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

Definition (Hybrid game α)

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

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

$$p(e_1,\dots,e_n) \mid e \geq \tilde{e} \mid \neg P \mid P \wedge Q \mid \forall x\, P \mid \exists x\, P \mid \langle\alpha\rangle P \mid [\alpha]P$$

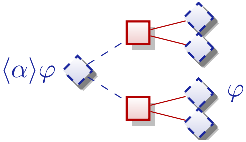
All Reals | Some Reals | Angel Wins | Demon Wins

TOCL'15

Summary

differential game logic

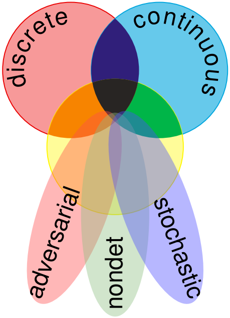
$$\mathrm{d}\mathcal{G}\mathcal{L} = \mathrm{GL} + \mathrm{HG} = \mathrm{d}\mathcal{L} + {}^{d}$$



- Differential game logic
- Logic for hybrid games
- Compositional PL + logic
- Discrete + continuous + adversarial
- Operational semantics (informally)

Next lecture

1. Formal semantics



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`.

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

André Platzer.
Logics of dynamical systems.
In LICS [10], 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.
The complete proof theory of hybrid systems.
In LICS [10], 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.

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.

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