

03: Choice & Control

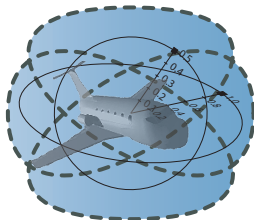
15-424: Foundations of Cyber-Physical Systems

André Platzer

aplatzer@cs.cmu.edu

Computer Science Department

Carnegie Mellon University, Pittsburgh, PA



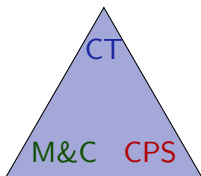
- 1 Learning Objectives
- 2 Gradual Introduction to Hybrid Programs
- 3 Notational Convention
- 4 Semantics of Hybrid Programs

- 1 Learning Objectives
- 2 Gradual Introduction to Hybrid Programs
- 3 Notational Convention
- 4 Semantics of Hybrid Programs

Learning Objectives

Choice & Control

nondeterminism
abstraction
programming languages for CPS
semantics
compositionality



models
core principles
discrete+
continuous

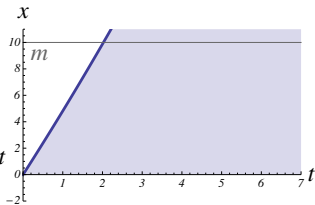
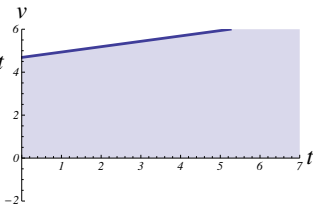
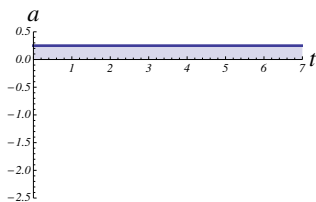
operational effect
operational precision

- 1 Learning Objectives
- 2 Gradual Introduction to Hybrid Programs**
- 3 Notational Convention
- 4 Semantics of Hybrid Programs

Playing with Acceleration and Braking

Example (Speedy the point)

$$a := a + 1; \{x' = v, v' = a\}$$



Playing with Acceleration and Braking

Example (Speedy the point)

$$a := -2; \{x' = v, v' = a\};$$

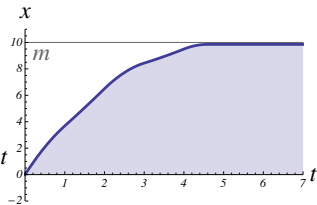
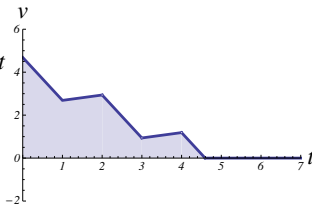
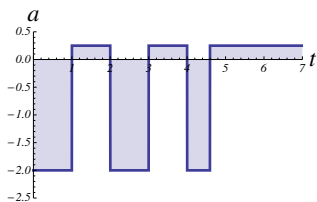
$$a := 0.25; \{x' = v, v' = a\};$$

$$a := -2; \{x' = v, v' = a\};$$

$$a := 0.25; \{x' = v, v' = a\};$$

$$a := -2; \{x' = v, v' = a\};$$

$$a := 0.25; \{x' = v, v' = a\}$$



- 1 Learning Objectives
- 2 Gradual Introduction to Hybrid Programs
- 3 Notational Convention**
- 4 Semantics of Hybrid Programs

Example (Naming Conventions)

Letters	Convention
x, y, z	variables
e, \tilde{e}	terms
P, Q	formulas
α, β	programs
c	constant symbols
f, g, h	function symbols
p, q, r	predicate symbols

In CPS applications, all bets are off because names follow application:
 x position v velocity and a acceleration variables

Notational Conventions: Precedence

Convention (Operator Precedence)

- 1 Unary operators (including $*$, \neg and $\forall x, \exists x$) bind stronger than binary.
- 2 \wedge bind stronger than \vee , which binds stronger than $\rightarrow, \leftrightarrow$
- 3 $;$ bind stronger than \cup
- 4 Arithmetic operators $+, -, \cdot$ associate to the left
- 5 Logical and program operators associate to the right

Example (Operator Precedence)

$$\forall x P \wedge Q \equiv (\forall x P) \wedge Q$$

$$\forall x P \rightarrow Q \equiv (\forall x P) \rightarrow Q.$$

$$\alpha; \beta \cup \gamma \equiv (\alpha; \beta) \cup \gamma$$

$$\alpha \cup \beta; \gamma \equiv \alpha \cup (\beta; \gamma)$$

$$\alpha; \beta^* \equiv \alpha; (\beta^*)$$

$$P \rightarrow Q \rightarrow R \equiv P \rightarrow (Q \rightarrow R).$$

But $\rightarrow, \leftrightarrow$ expect explicit parentheses. Illegal: $P \rightarrow Q \leftrightarrow R$ $P \leftrightarrow Q \rightarrow R$

- 1 Learning Objectives
- 2 Gradual Introduction to Hybrid Programs
- 3 Notational Convention
- 4 Semantics of Hybrid Programs**

Definition (Hybrid program semantics)

$([\![\cdot]\!] : \text{HP} \rightarrow \wp(\mathcal{S} \times \mathcal{S}))$

$$[\![x := e]\!] = \{(\omega, \nu) : \nu = \omega \text{ except } \nu[x] = \omega[e]\}$$

$$[\![?Q]\!] = \{(\omega, \omega) : \omega \in [\![Q]\!]\}$$

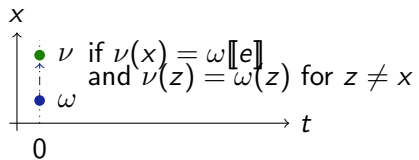
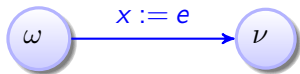
$$[\![x' = f(x)]\!] = \{(\varphi(0), \varphi(r)) : \varphi \models x' = f(x) \text{ for some duration } r\}$$

$$[\![\alpha \cup \beta]\!] = [\![\alpha]\!] \cup [\![\beta]\!]$$

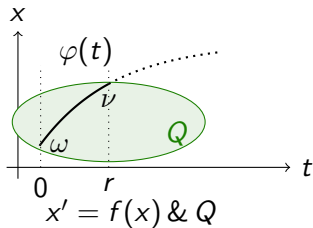
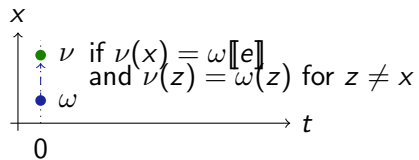
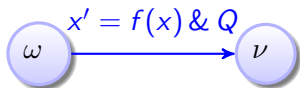
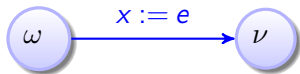
$$[\![\alpha; \beta]\!] = [\![\alpha]\!] \circ [\![\beta]\!]$$

$$[\![\alpha^*]\!] = \bigcup_{n \in \mathbb{N}} [\![\alpha^n]\!]$$

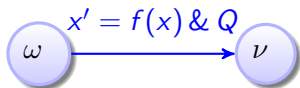
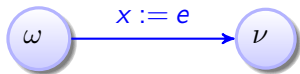
Hybrid Program: Semantics



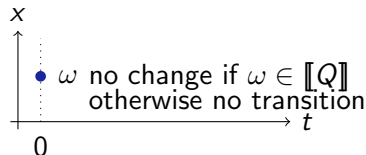
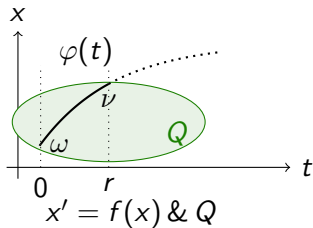
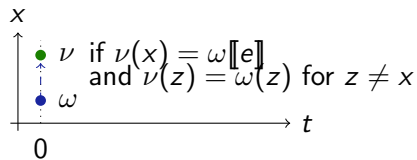
Hybrid Program: Semantics



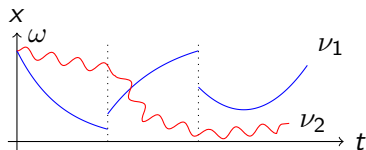
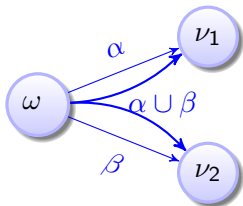
Hybrid Program: Semantics



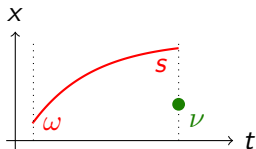
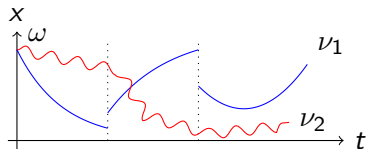
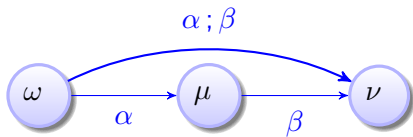
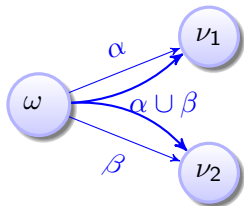
if $\omega \in \llbracket Q \rrbracket$



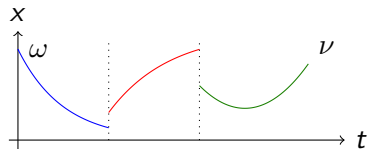
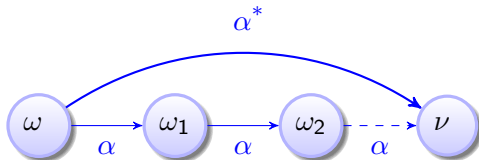
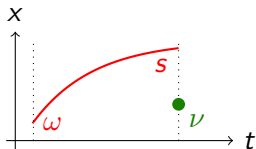
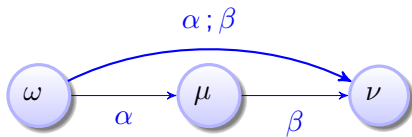
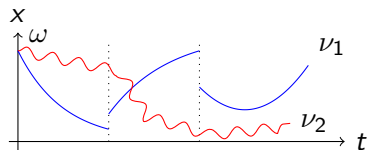
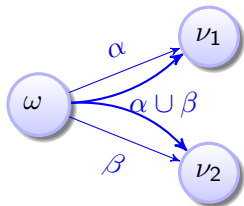
Hybrid Program: Semantics



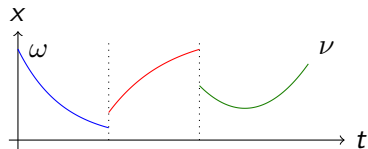
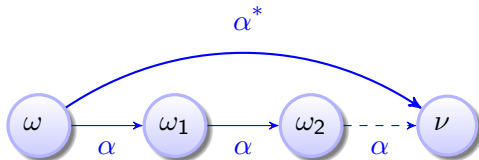
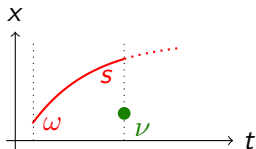
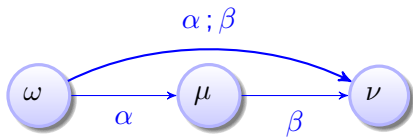
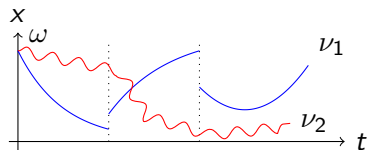
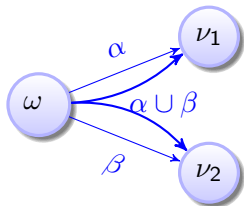
Hybrid Program: Semantics



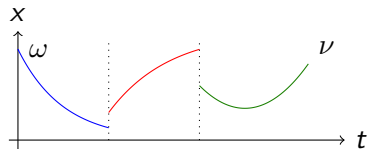
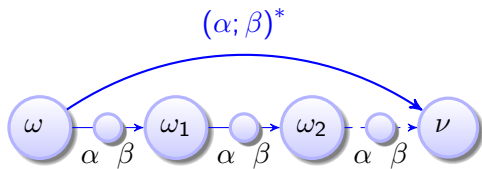
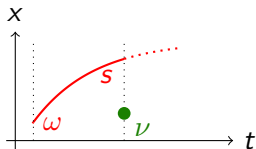
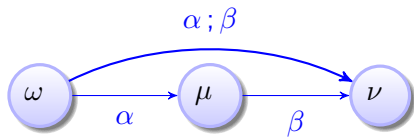
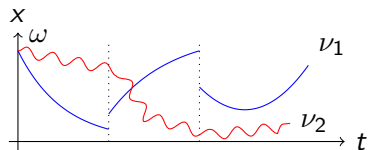
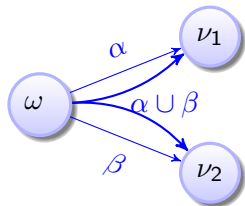
Hybrid Program: Semantics



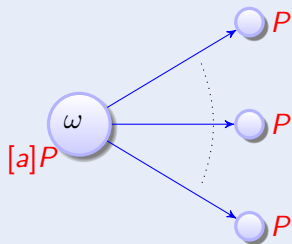
Hybrid Program: Semantics



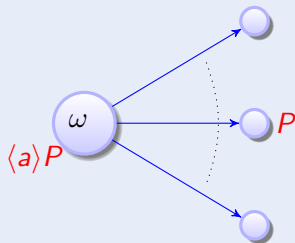
Hybrid Program: Semantics



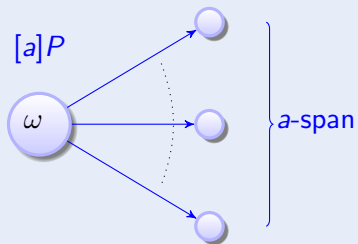
Definition (dL Formulas)



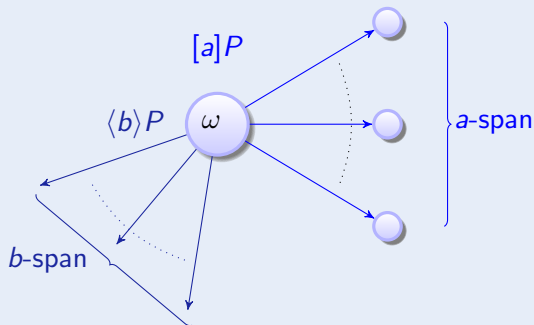
Definition (dL Formulas)



Definition (dL Formulas)

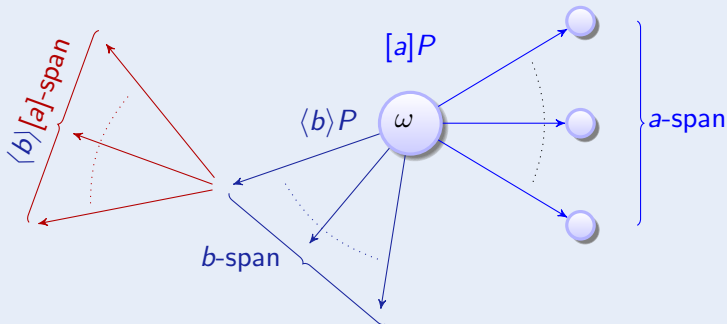


Definition (dL Formulas)

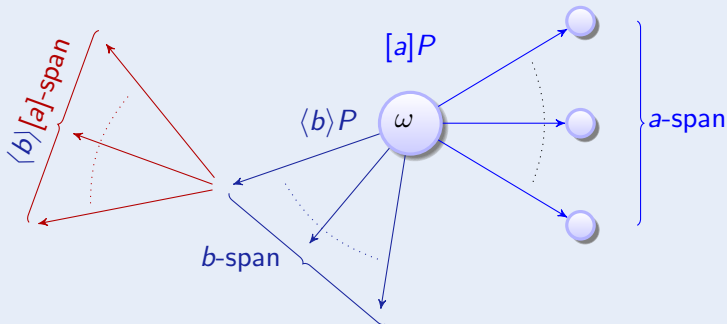


Hybrid Program: Semantics

Definition (dL Formulas)



Definition (dL Formulas)



compositional semantics \Rightarrow compositional proofs!

Definition (Hybrid program semantics)

$([\cdot] : \text{HP} \rightarrow \wp(\mathcal{S} \times \mathcal{S}))$

$$[[x := e]] = \{(\omega, \nu) : \nu = \omega \text{ except } \nu[x] = \omega[e]\}$$

$$[[?Q]] = \{(\omega, \omega) : \omega \in [[Q]]\}$$

$$[[x' = f(x)]] = \{(\varphi(0), \varphi(r)) : \varphi \models x' = f(x) \text{ for some duration } r\}$$

$$[[\alpha \cup \beta]] = [[\alpha]] \cup [[\beta]]$$

$$[[\alpha; \beta]] = [[\alpha]] \circ [[\beta]]$$

$$[[\alpha^*]] = \bigcup_{n \in \mathbb{N}} [[\alpha^n]]$$