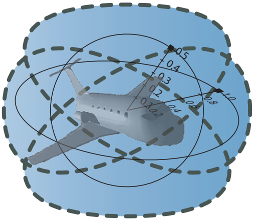
Logical Analysis of Hybrid Systems

A Complete Answer to a Complexity Challenge

André Platzer

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

http://symbolaris.com/

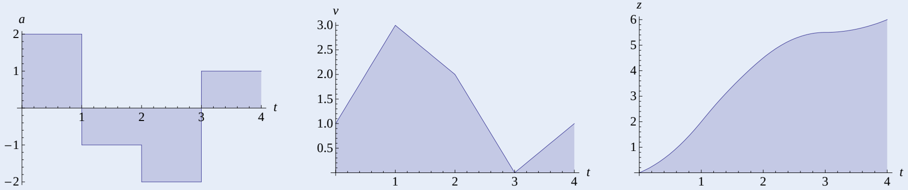


How can we design computers that are guaranteed to interact correctly with the physical world?

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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



Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

1. More than computers: no `NullPointerException` $\not\Rightarrow$ safe

Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

1. More than computers: no `NullPointerException` $\not\Rightarrow$ safe
2. More than physics: braking control $v^2 \leq 2b(M - z) \not\Rightarrow$ safe

Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

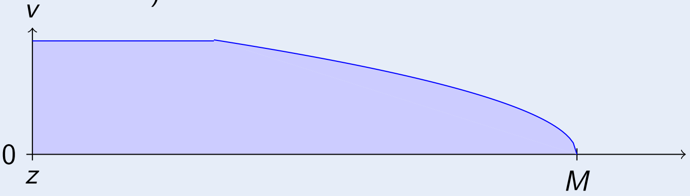
1. More than computers: no `NullPointerException` $\not\Rightarrow$ safe
2. More than physics: braking control $v^2 \le 2b(M-z) \not\Rightarrow$ safe
3. Joint dynamics requires: $SB \ge \frac{v^2}{2b} + \frac{a^2\varepsilon^2}{2b} + \frac{a}{b}\varepsilon v + \frac{a}{2}\varepsilon^2 + \varepsilon v \ldots$

Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

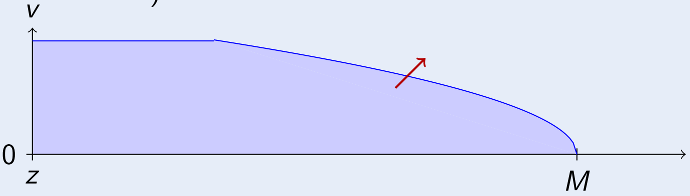


Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

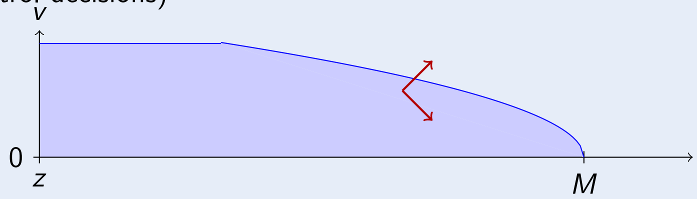


Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

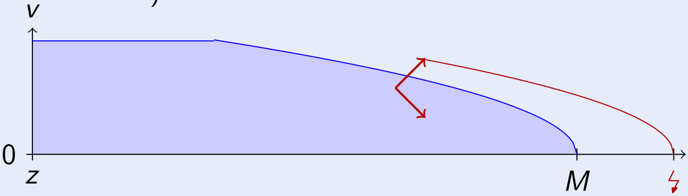


Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

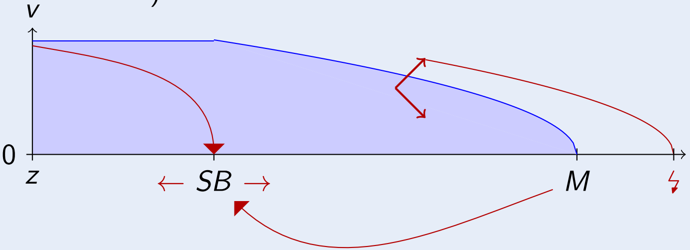


Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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

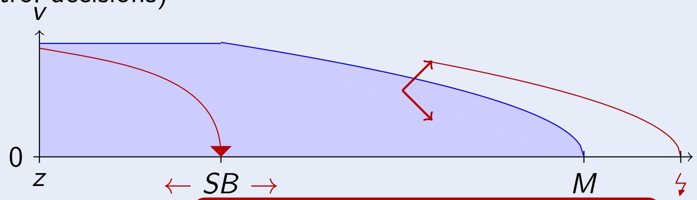


Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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



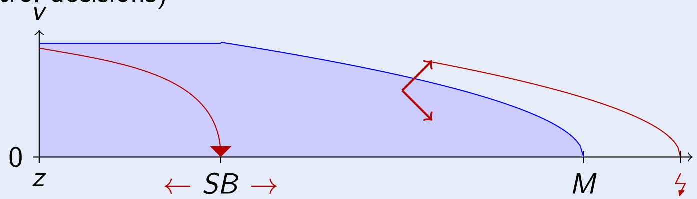
$$SB \geq \frac{v^2}{2b} + \frac{a^2\varepsilon^2}{2b} + \frac{a}{b}\varepsilon v + \frac{a}{2}\varepsilon^2 + \varepsilon v$$

Hybrid Systems Analysis: Car Control

Challenge (Hybrid Systems)

Fixed rule describing state evolution with both

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



$\forall M \, \exists SB$ "Car always safe"

Hybrid Systems Analysis: Car Control

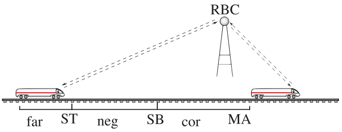
Challenge (Hybrid Systems)

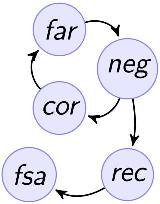
Fixed rule describing state evolution with both

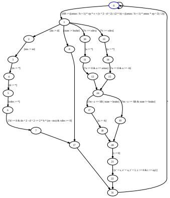
- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)

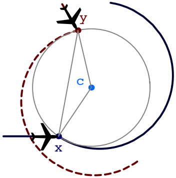
"One law to rule them all, and in the darkness bind them"

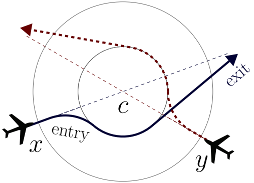
Successful Hybrid Systems Proofs

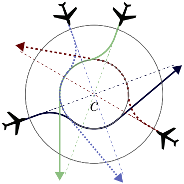


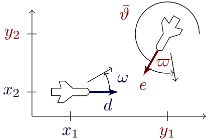


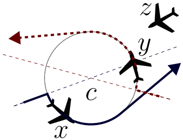


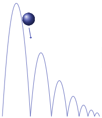


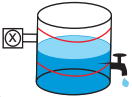




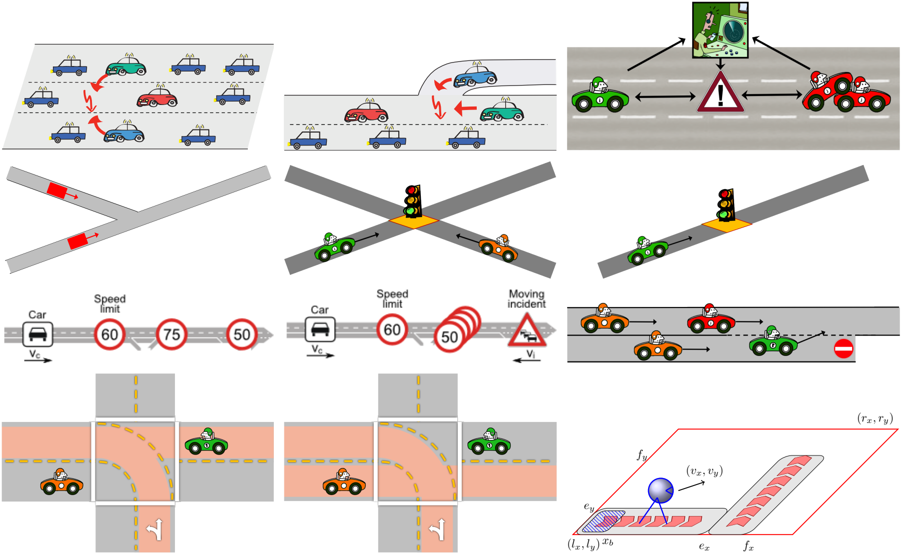








Successful Hybrid Systems Proofs



Differential Dynamic Logic for Hybrid Systems

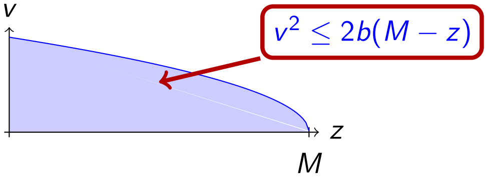
differential dynamic logic

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

Differential Dynamic Logic for Hybrid Systems

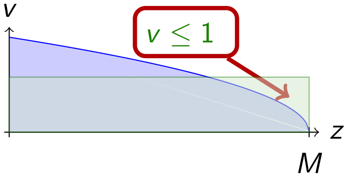
differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}}$



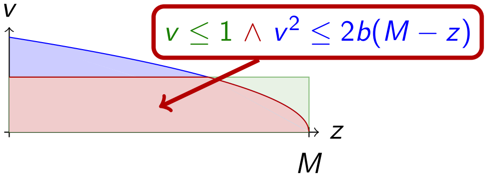
differential dynamic logic

$\mathrm{d}\mathcal{L} = \mathrm{FOL}_{\mathbb{R}}$



differential dynamic logic

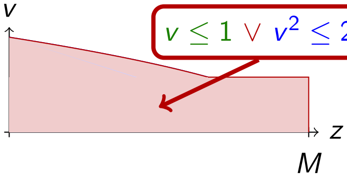
$d\mathcal{L} = \text{FOL}_{\mathbb{R}}$



Differential Dynamic Logic for Hybrid Systems

differential dynamic logic

$d\mathcal{L} = \text{FOL}_{\mathbb{R}}$



$v \leq 1 \lor v^2 \leq 2b(M - z)$

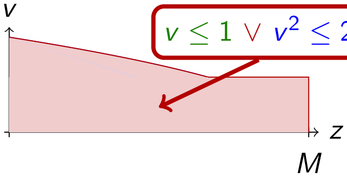
Differential Dynamic Logic for Hybrid Systems

differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}}$

$\forall M \, \exists SB \, \ldots$

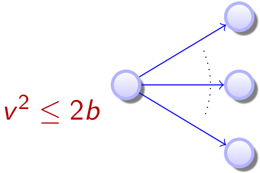
$\forall t \geq 0 \, \ldots$



$v \leq 1 \lor v^2 \leq 2b(M - z)$

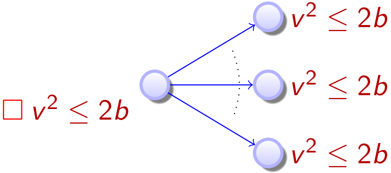
differential dynamic logic

$d\mathcal{L} = \text{FOL}_{\mathbb{R}} +$



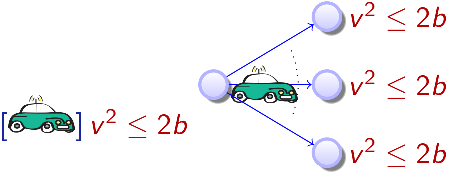
differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{ML}$



differential dynamic logic

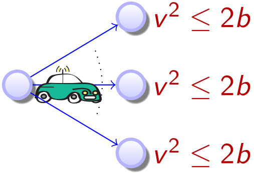
$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL}$



differential dynamic logic

$d\mathcal{L} = \text{FOL}_{\mathbb{R}} + \text{DL} + \text{HP}$

$[z'' = a]\, v^2 \leq 2b$

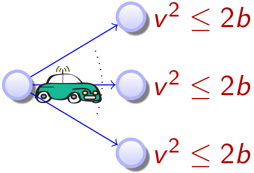


Differential Dynamic Logic for Hybrid Systems

differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL} + \mathrm{HP}$

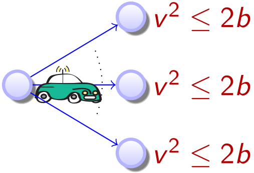
$$[\mathtt{if}(z > SB)\, a := -b;\ z'' = a]\, v^2 \leq 2b$$



differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL} + \mathrm{HP}$

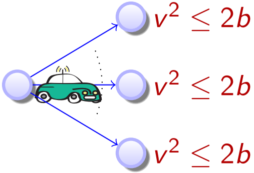
$$[\underbrace{\mathtt{if}(z > SB)\, a := -b;\ z'' = a}_{\text{hybrid program}}]\, v^2 \leq 2b$$



differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL} + \mathrm{HP}$

$$\mathcal{C} \to [\underbrace{\mathtt{if}(z > SB)\, a := -b;\ z'' = a}_{\text{hybrid program}}]\, v^2 \leq 2b$$

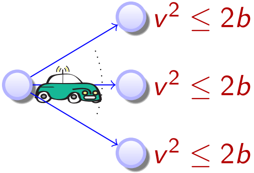


Differential Dynamic Logic for Hybrid Systems

differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL} + \mathrm{HP}$

$$\mathcal{C} \rightarrow [\underbrace{\mathtt{if}(z > SB)\, a := -b;\;\; z'' = a}_{\text{hybrid program}}]\, v^2 \leq 2b$$



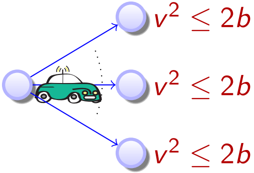
Initial condition

Differential Dynamic Logic for Hybrid Systems

differential dynamic logic

$d\mathcal{L} = \text{FOL}_{\mathbb{R}} + \text{DL} + \text{HP}$

$$\mathcal{C} \rightarrow [\underbrace{\texttt{if}(z > SB)\, a := -b;\ z'' = a}_{\text{hybrid program}}]\, v^2 \le 2b$$



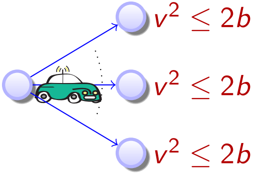
Initial condition

System dynamics

differential dynamic logic

$d\mathcal{L} = \mathrm{FOL}_{\mathbb{R}} + \mathrm{DL} + \mathrm{HP}$

$$\mathcal{C} \rightarrow [\underbrace{\mathtt{if}(z > SB)\, a := -b;\ z'' = a}_{\text{hybrid program}}]\, v^2 \leq 2b$$



Initial condition

System dynamics

Post condition

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

Definition (Hybrid program α)

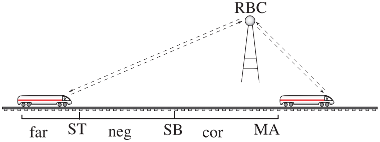
$x' = f(x)$	(continuous evolution)	
$x := f(x)$	(discrete jump)	jump & test
$?H$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

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

Definition (Hybrid program α)

$x' = f(x)$	(continuous evolution)	
$x := f(x)$	(discrete jump)	jump & test
$?H$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

$$train \equiv (ctrl; drive)^*$$
$$ctrl \equiv (?M - z \leq SB; a := -b)$$
$$\cup (?M - z \geq SB; a := \ldots)$$
$$drive \equiv z'' = a$$
$$\& v \geq 0 \land \tau \leq \varepsilon$$

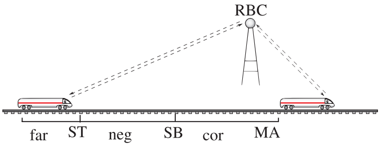


Differential Dynamic Logic $d\mathcal{L}$: Syntax

Definition (Hybrid program α)

$x' = f(x)$	(continuous evolution)	
$x := f(x)$	(discrete jump)	jump & test
$?H$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

$$
\begin{aligned}
train &\equiv (ctrl; drive)^* \\
ctrl &\equiv (?M - z \leq SB; a := -b) \\
&\quad \cup (?M - z \geq SB; a := \ldots) \\
drive &\equiv \tau := 0; z' = v, v' = a, \tau' = 1 \\
&\quad \& \, v \geq 0 \land \tau \leq \varepsilon
\end{aligned}
$$



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

Definition (Hybrid program α)

$x' = f(x) \,\&\, H$	(continuous evolution)	
$x := f(x)$	(discrete jump)	jump & test
$?H$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

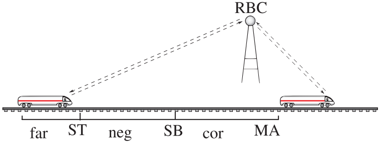
$$train \equiv (ctrl; drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

$$\cup\, (?M - z \geq SB; a := \ldots)$$

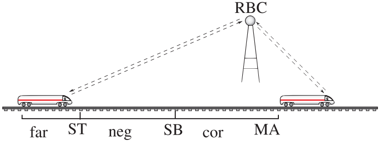
$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\&\, v \geq 0 \land \tau \leq \varepsilon$$

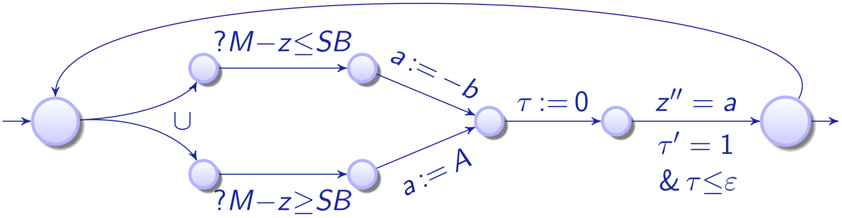


Branching Transitions in Hybrid Programs

$$
\begin{aligned}
train &\equiv (ctrl\,; drive)^* \\
ctrl &\equiv (?M - z \leq SB; a := -b) \\
&\quad \cup (?M - z \geq SB; a := A) \\
drive &\equiv \tau := 0; z' = v, v' = a, \tau' = 1 \\
&\quad \& \, v \geq 0 \land \tau \leq \varepsilon
\end{aligned}
$$



Branching Transitions in Hybrid Programs



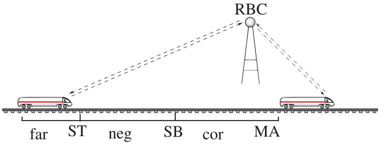
$$train \equiv (ctrl; drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

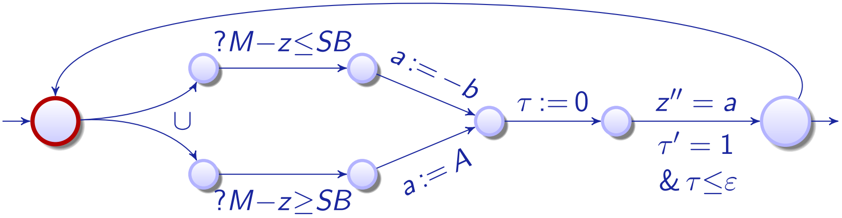
$$\cup (?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\& \, v \geq 0 \wedge \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



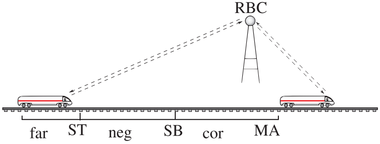
$$train \equiv (ctrl; drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

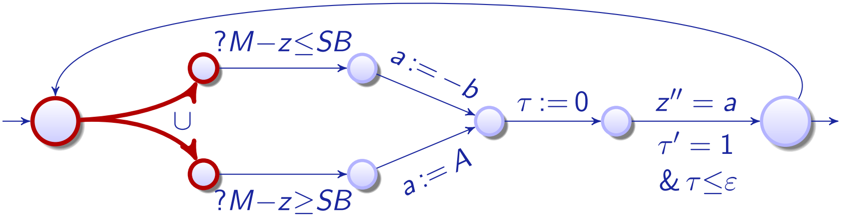
$$\cup (?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

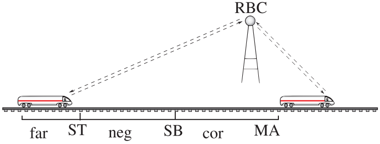
$$\& \, v \geq 0 \land \tau \leq \varepsilon$$



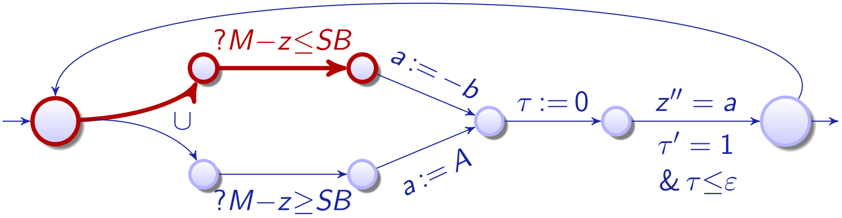
Branching Transitions in Hybrid Programs



$$train \equiv (\textcolor{red}{ctrl};\, drive)^*$$
$$ctrl \equiv (?M - z \leq SB;\, a := -b)$$
$$\textcolor{red}{\cup}\, (?M - z \geq SB;\, a := A)$$
$$drive \equiv \tau := 0;\, z' = v, v' = a, \tau' = 1$$
$$\&\, v \geq 0 \land \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



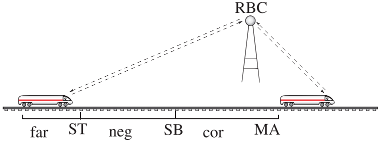
$$train \equiv (ctrl; drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

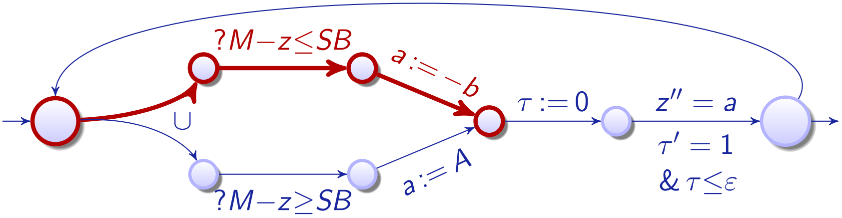
$$\cup (?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\& v \geq 0 \land \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



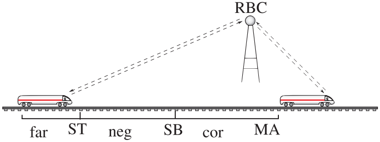
$$train \equiv (ctrl;\, drive)^*$$

$$ctrl \equiv (?M - z \leq SB;\, a := -b)$$

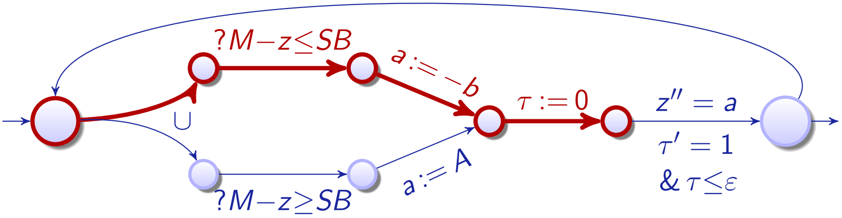
$$\cup\, (?M - z \geq SB;\, a := A)$$

$$drive \equiv \tau := 0;\, z' = v,\, v' = a,\, \tau' = 1$$

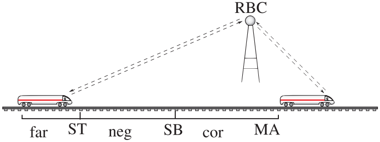
$$\&\, v \geq 0 \land \tau \leq \varepsilon$$



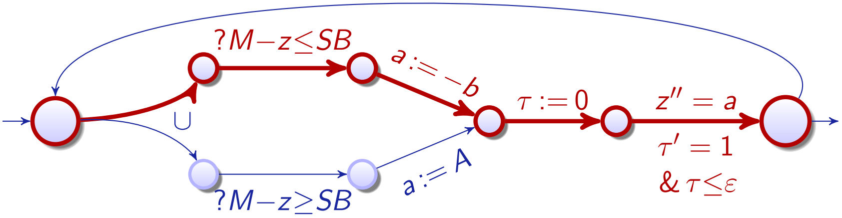
Branching Transitions in Hybrid Programs



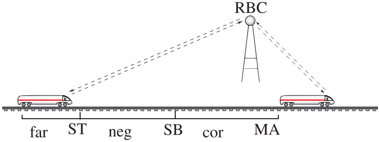
$$train \equiv (ctrl\,;\, drive)^*$$
$$ctrl \equiv (?M - z \leq SB; a := -b)$$
$$\cup\, (?M - z \geq SB; a := A)$$
$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$
$$\&\ v \geq 0 \wedge \tau \leq \varepsilon$$



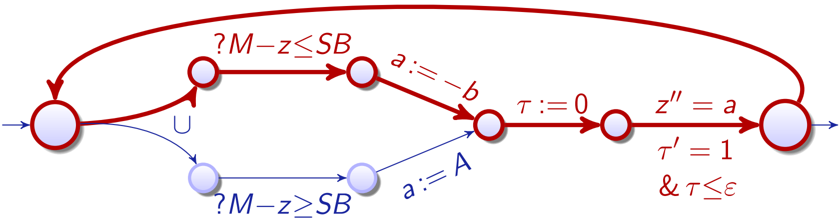
Branching Transitions in Hybrid Programs



$$
\begin{aligned}
train &\equiv (ctrl\,;\, drive)^* \\
ctrl &\equiv (?M - z \leq SB;\, a := -b) \\
&\quad \cup (?M - z \geq SB;\, a := A) \\
drive &\equiv \tau := 0;\, z' = v, v' = a, \tau' = 1 \\
&\quad \& \; v \geq 0 \land \tau \leq \varepsilon
\end{aligned}
$$



Branching Transitions in Hybrid Programs



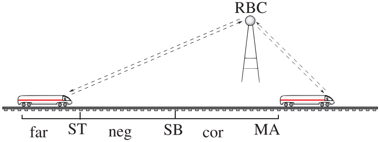
$$train \equiv (ctrl\,;drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

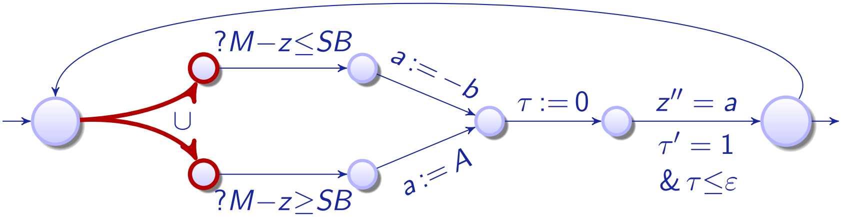
$$\cup\,(?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

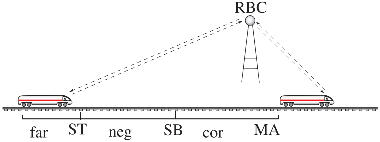
$$\&\, v \geq 0 \wedge \tau \leq \varepsilon$$



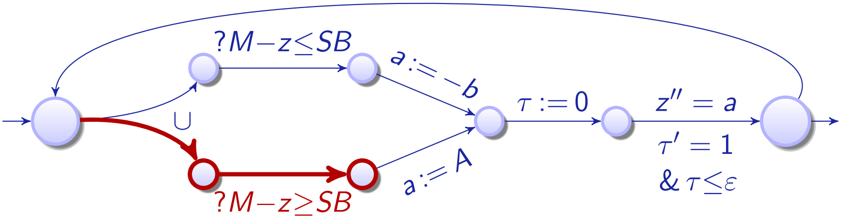
Branching Transitions in Hybrid Programs



$$
\begin{aligned}
train &\equiv (\color{red}{ctrl}\color{black}; drive)^* \\
ctrl &\equiv (?M - z \le SB; a := -b) \\
&\quad \color{red}{\cup}\color{black}\ (?M - z \ge SB; a := A) \\
drive &\equiv \tau := 0; z' = v, v' = a, \tau' = 1 \\
&\quad \& \ v \ge 0 \land \tau \le \varepsilon
\end{aligned}
$$



Branching Transitions in Hybrid Programs



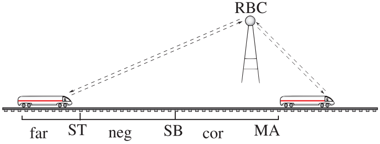
$$train \equiv (ctrl; drive)^*$$

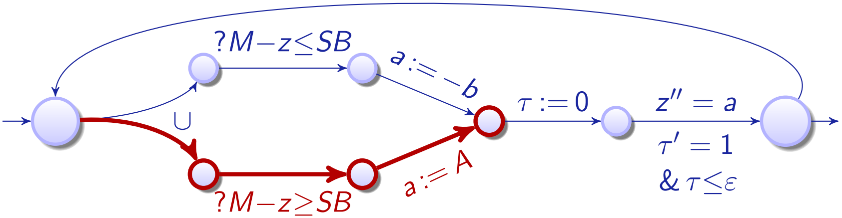
$$ctrl \equiv (?M - z \le SB; a := -b)$$

$$\cup\, (?M - z \ge SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\& \, v \ge 0 \land \tau \le \varepsilon$$





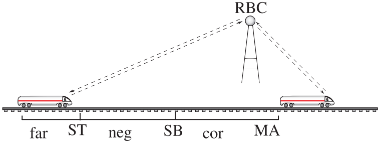
$$train \equiv (ctrl; drive)^*$$

$$ctrl \equiv (?M - z \leq SB; a := -b)$$

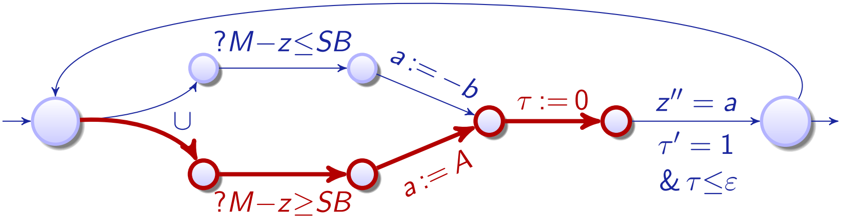
$$\cup (?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\& \; v \geq 0 \land \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



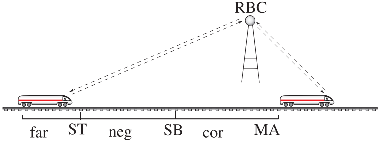
$$train \equiv (ctrl\,;\, drive)^*$$

$$ctrl \equiv (?M - z \leq SB;\, a := -b)$$

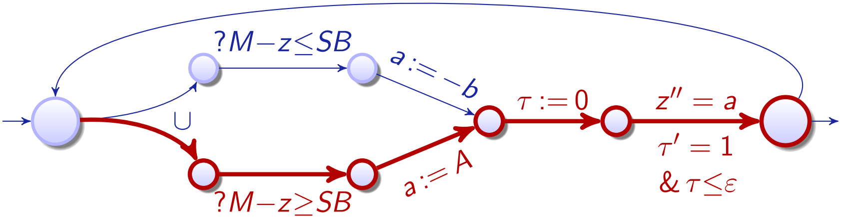
$$\cup\, (?M - z \geq SB;\, a := A)$$

$$drive \equiv \tau := 0;\, z' = v,\, v' = a,\, \tau' = 1$$

$$\&\, v \geq 0 \wedge \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



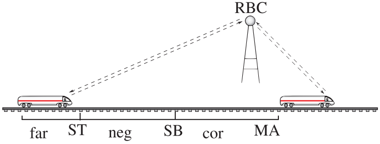
$$train \equiv (ctrl\,;\, drive)^*$$

$$ctrl \equiv (?M - z \leq SB;\, a := -b)$$

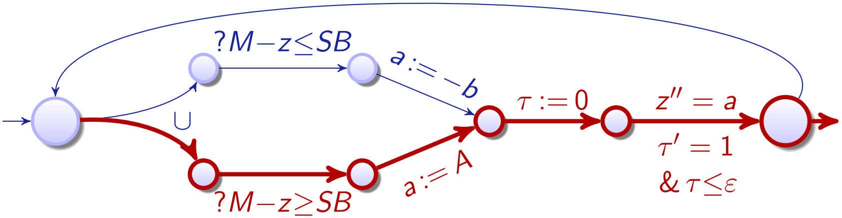
$$\cup\, (?M - z \geq SB;\, a := A)$$

$$drive \equiv \tau := 0;\, z' = v,\, v' = a,\, \tau' = 1$$

$$\&\, v \geq 0 \land \tau \leq \varepsilon$$



Branching Transitions in Hybrid Programs



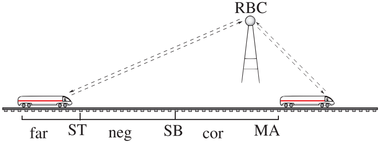
$$train \equiv (ctrl\,;drive)^*$$

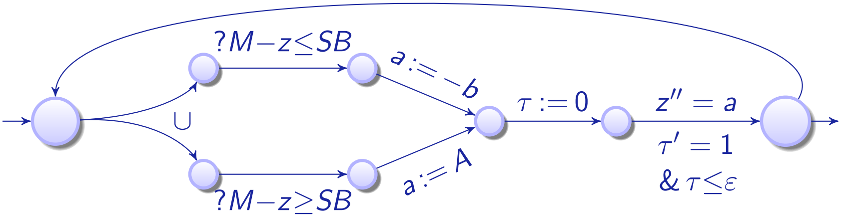
$$ctrl \equiv (?M - z \leq SB; a := -b)$$

$$\cup\, (?M - z \geq SB; a := A)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$

$$\&\, v \geq 0 \land \tau \leq \varepsilon$$

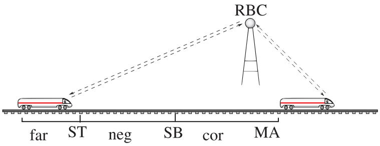




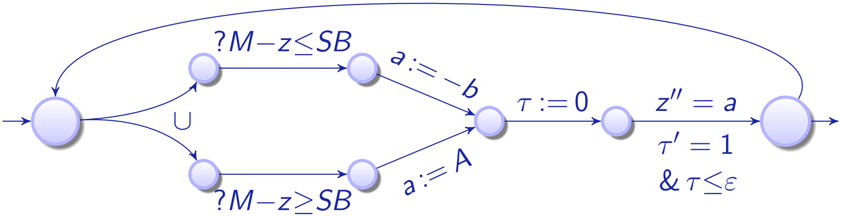
$\mathtt{if}(H)\,\alpha\,\mathtt{else}\,\beta \equiv$

$\mathtt{while}(H)\,\alpha \equiv$

$$train \equiv (ctrl\,;\,drive)^*$$
$$ctrl \equiv (?M - z \leq SB;\, a := -b)$$
$$\cup\, (?M - z \geq SB;\, a := A)$$
$$drive \equiv \tau := 0;\, z' = v, v' = a, \tau' = 1$$
$$\&\, v \geq 0 \land \tau \leq \varepsilon$$

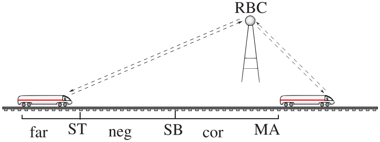


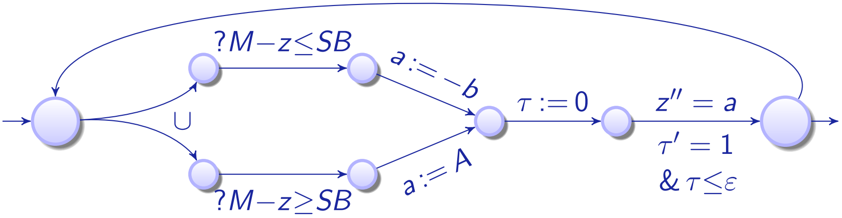
Branching Transitions in Hybrid Programs



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

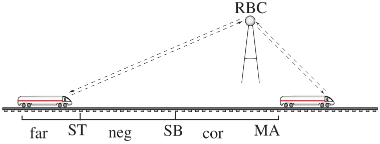
$$train \equiv (ctrl\,;drive)^*$$
$$ctrl \equiv (?M - z \le SB; a := -b)$$
$$\cup\,(?M - z \ge SB; a := A)$$
$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1$$
$$\&\, v \ge 0 \land \tau \le \varepsilon$$





$$\texttt{if}(H)\,\alpha\,\texttt{else}\,\beta \equiv (?H;\alpha)\cup(?\neg H;\beta)$$
$$\texttt{while}(H)\,\alpha \equiv (?H;\alpha)^*;?\neg H$$

$$\begin{aligned}
\mathit{train} &\equiv (\mathit{ctrl};\mathit{drive})^*\\
\mathit{ctrl} &\equiv (?M-z\leq SB;\,a:=-b)\\
&\quad\cup(?M-z\geq SB;\,a:=A)\\
\mathit{drive} &\equiv \tau:=0;\,z'=v,v'=a,\tau'=1\\
&\quad\&\,v\geq 0\land\tau\leq\varepsilon
\end{aligned}$$



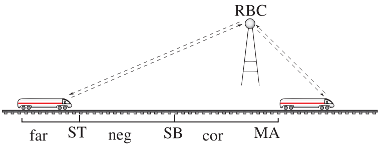
Differential Dynamic Logic d$\mathcal{L}$: Syntax

Definition (d$\mathcal{L}$ Formula ϕ)

$$\theta_1 \geq \theta_2 \mid \neg\phi \mid \phi \wedge \psi \mid \phi \vee \psi \mid \phi \rightarrow \psi \mid \forall x\, \phi \mid \exists x\, \phi \mid [\alpha]\phi \mid \langle\alpha\rangle\phi$$

with terms θ_1, θ_2 of nonlinear real arithmetic $(+, \cdot)$

$$SB \geq \ldots \;\rightarrow\; [(ctrl\,;drive)^*]\, z \leq M$$



All trains respect M
RBC partitions M
$\Rightarrow$ system collision free

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

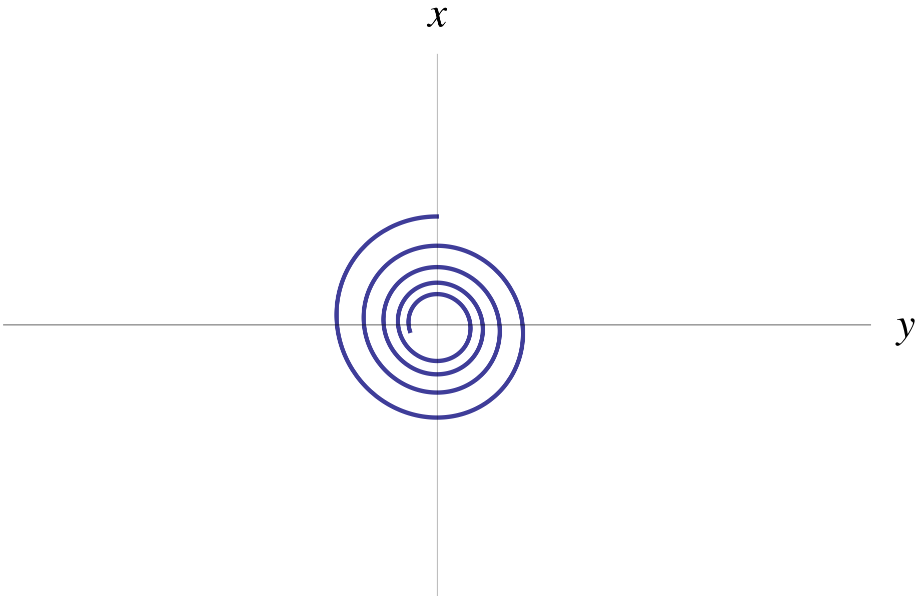
Definition (Hybrid program α)

$$
\begin{aligned}
\rho(x := \theta) &= \{(v, w) : w = v \text{ except } [\![x]\!]_w = [\![\theta]\!]_v\} \\
\rho(?H) &= \{(v, v) : v \models H\} \\
\rho(x' = f(x)) &= \{(\varphi(0), \varphi(r)) : \varphi \models x' = f(x) \text{ for some duration } r\} \\
\rho(\alpha \cup \beta) &= \rho(\alpha) \cup \rho(\beta) \\
\rho(\alpha; \beta) &= \rho(\beta) \circ \rho(\alpha) \\
\rho(\alpha^*) &= \bigcup_{n \in \mathbb{N}} \rho(\alpha^n)
\end{aligned}
$$

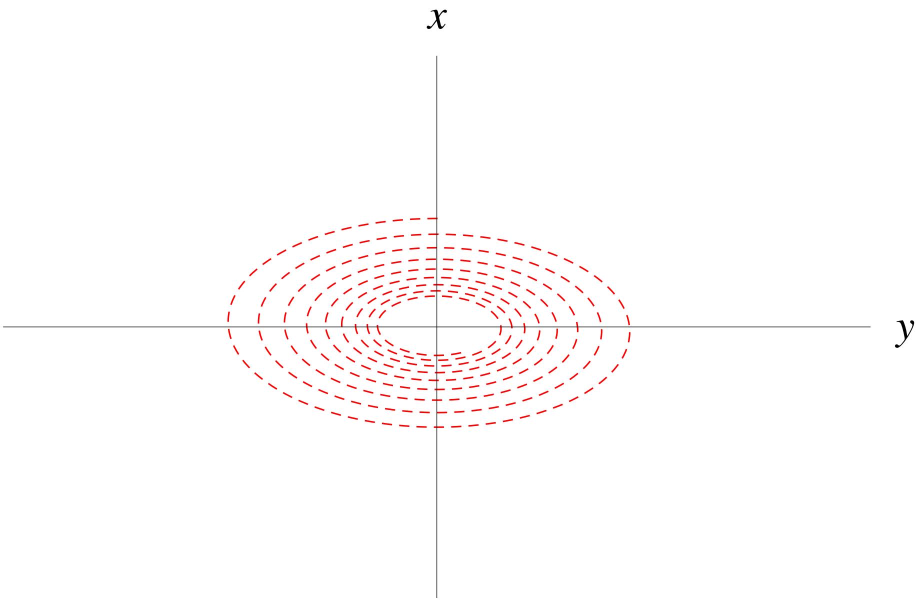
Definition ($d\mathcal{L}$ Formula ϕ)

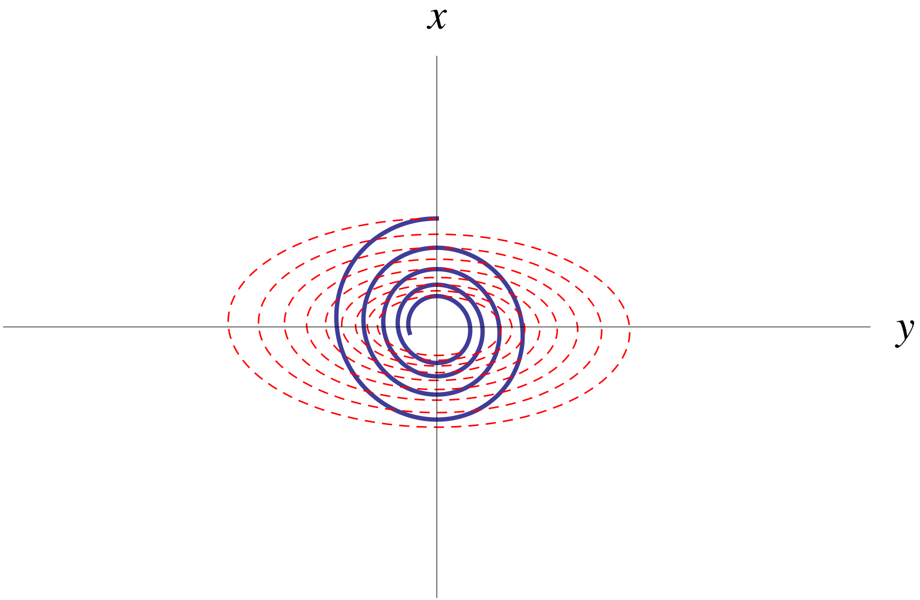
$v \models \theta_1 \geq \theta_2$	iff	$[\![\theta_1]\!]_v \geq [\![\theta_2]\!]_v$
$v \models [\alpha]\phi$	iff	$w \models \phi$ for all w with $v\rho(\alpha)w$
$v \models \langle\alpha\rangle\phi$	iff	$w \models \phi$ for some w with $v\rho(\alpha)w$
$v \models \forall x\, \phi$	iff	$w \models \phi$ for all w that agree with v except for x
$v \models \exists x\, \phi$	iff	$w \models \phi$ for some w that agrees with v except for x
$v \models \phi \land \psi$	iff	$v \models \phi$ and $v \models \psi$

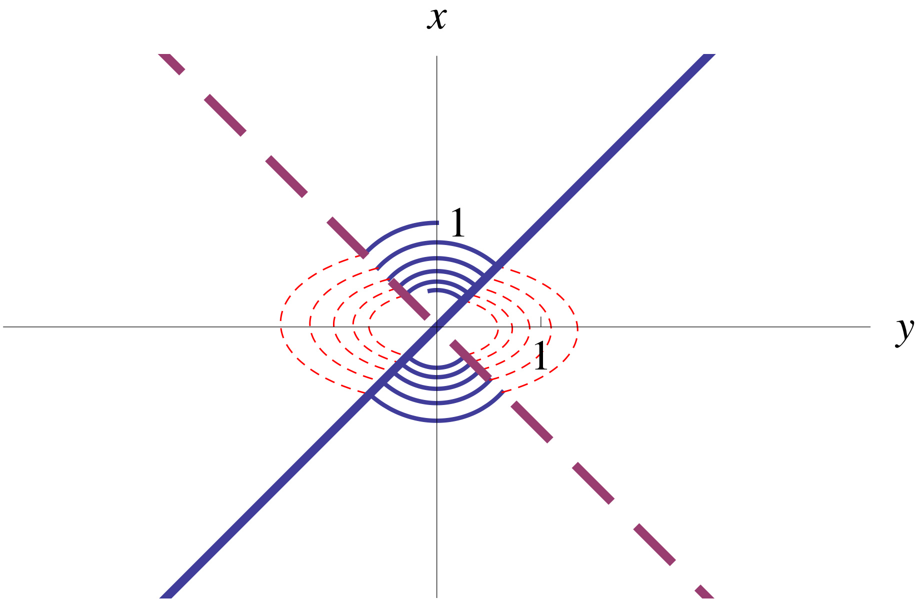
Compositionality in Hybrid Systems



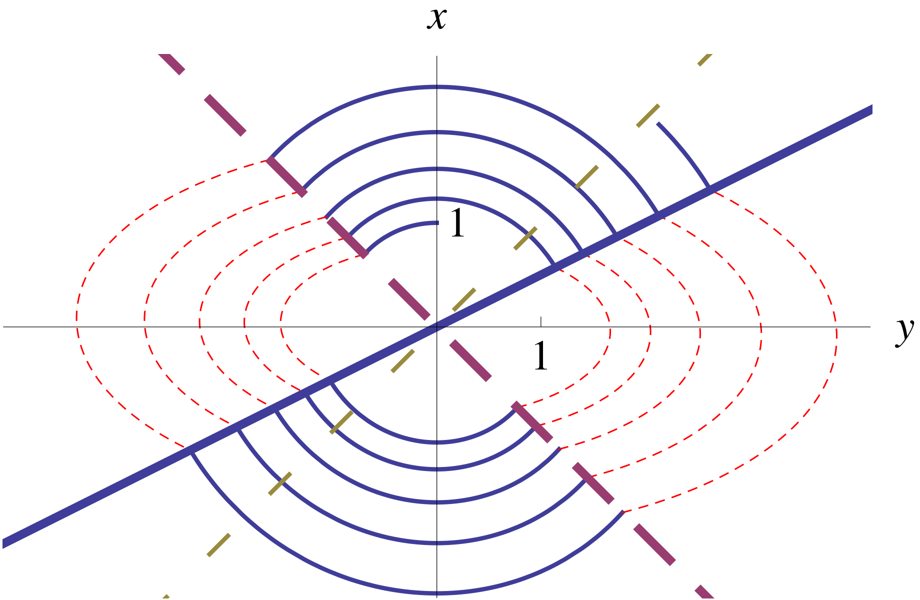
Compositionality in Hybrid Systems




x
y

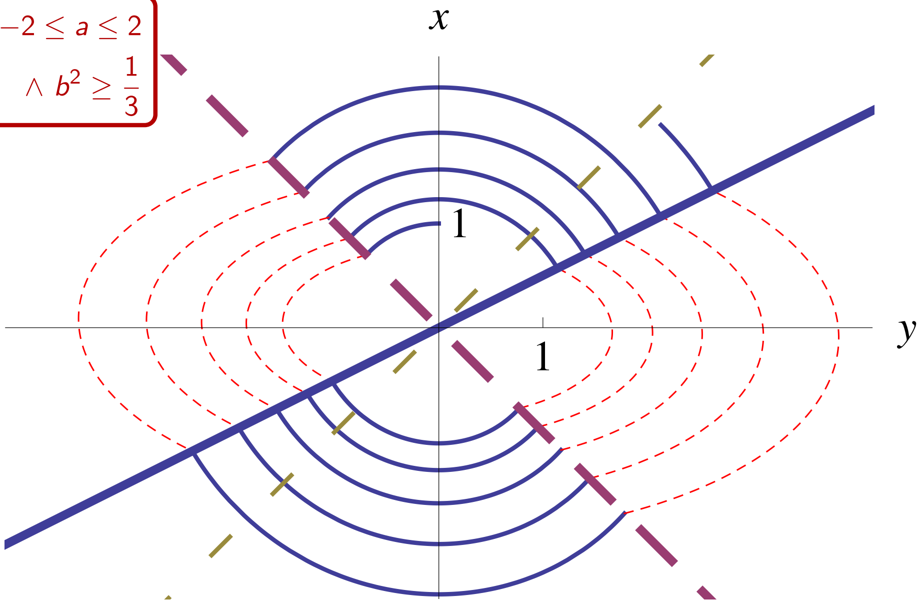

x
y
1
1

Compositionality in Hybrid Systems



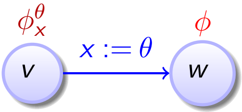
Compositionality in Hybrid Systems

$$-2 \leq a \leq 2 \land b^2 \geq \frac{1}{3}$$



Proof Calculus for Differential Dynamic Logic

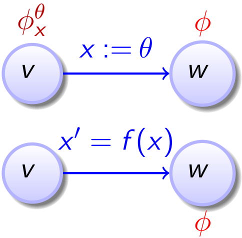
$$\frac{\phi_x^\theta}{\langle x := \theta \rangle \phi}$$



Proof Calculus for Differential Dynamic Logic

$$\frac{\phi_x^\theta}{\langle x := \theta \rangle \phi}$$

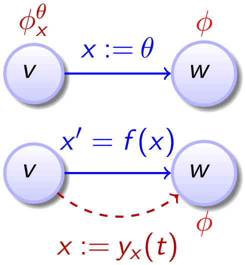
$$\frac{\exists t \geq 0\, \langle x := y_x(t) \rangle \phi}{\langle x' = f(x) \rangle \phi}$$



Proof Calculus for Differential Dynamic Logic

$$\frac{\phi_x^\theta}{\langle x := \theta\rangle\phi}$$

$$\frac{\exists t{\geq}0\,\langle x := y_x(t)\rangle\phi}{\langle x' = f(x)\rangle\phi}$$

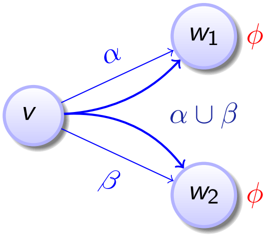


Proof Calculus for Differential Dynamic Logic

compositional semantics $\Rightarrow$ compositional rules!

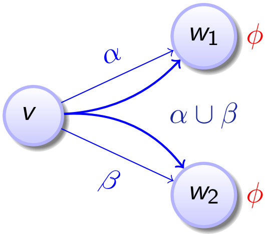
Proof Calculus for Differential Dynamic Logic

$$\frac{[\alpha]\phi \land [\beta]\phi}{[\alpha \cup \beta]\phi}$$

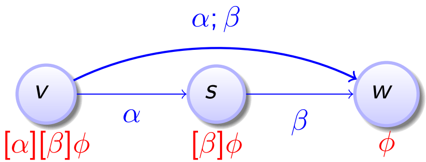


Proof Calculus for Differential Dynamic Logic

$$\frac{[\alpha]\phi \land [\beta]\phi}{[\alpha \cup \beta]\phi}$$

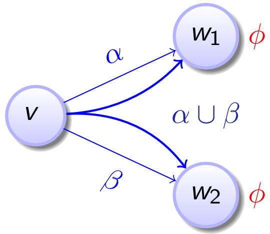


$$\frac{[\alpha][\beta]\phi}{[\alpha; \beta]\phi}$$

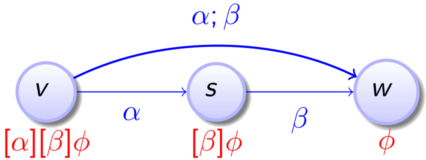


Proof Calculus for Differential Dynamic Logic

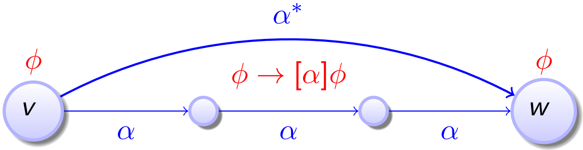
$$\frac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi}$$



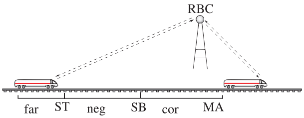
$$\frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi}$$



$$\frac{\phi \quad (\phi \rightarrow [\alpha]\phi)}{[\alpha^*]\phi}$$

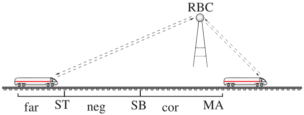


Deduction Modulo Real Arithmetic



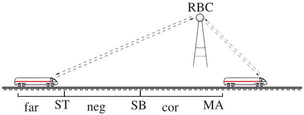
$$v \geq 0 \land z < M \rightarrow \langle z' = v, v' = -b \rangle \; z > M$$

Deduction Modulo Real Arithmetic



$$\dfrac{\dfrac{}{v \geq 0, z < M \rightarrow \exists t \geq 0 \, \langle z := -\frac{b}{2} t^2 + vt + z \rangle z > M}}{\dfrac{v \geq 0, z < M \rightarrow \langle z' = v, v' = -b \rangle z > M}{v \geq 0 \land z < M \rightarrow \langle z' = v, v' = -b \rangle \, z > M}}$$

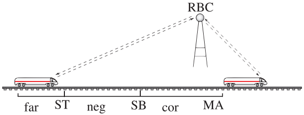
Deduction Modulo Real Arithmetic



Collins/Tarski QE not applicable!

$$\frac{\dfrac{v \geq 0, z < M \rightarrow \exists t{\geq}0\, \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}{v \geq 0, z < M \rightarrow \langle z' = v, v' = -b\rangle z > M}}{v \geq 0 \wedge z < M \rightarrow \langle z' = v, v' = -b\rangle\, z > M}$$

Deduction Modulo (Side Deduction)

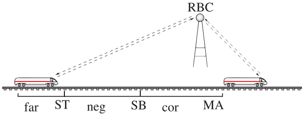


$$\frac{}{v \geq 0, z < M \rightarrow t \geq 0 \land \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}$$

start side

$$\cfrac{\cfrac{\overline{v \geq 0, z < M \rightarrow \exists t{\geq}0\, \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}}{v \geq 0, z < M \rightarrow \langle z' = v, v' = -b\rangle z > M}}{v \geq 0 \land z < M \rightarrow \langle z' = v, v' = -b\rangle\ z > M}$$

Deduction Modulo (Side Deduction)

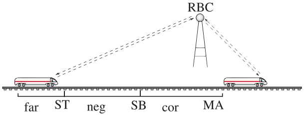


$$\dfrac{v \geq 0, z < M \rightarrow t \geq 0 \qquad \dfrac{v \geq 0, z < M \rightarrow -\frac{b}{2}t^2 + vt + z > M}{v \geq 0, z < M \rightarrow \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}}{v \geq 0, z < M \rightarrow t \geq 0 \land \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}$$

start side

$$\dfrac{\dfrac{}{v \geq 0, z < M \rightarrow \exists t \geq 0\, \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}}{\dfrac{v \geq 0, z < M \rightarrow \langle z' = v, v' = -b\rangle z > M}{v \geq 0 \land z < M \rightarrow \langle z' = v, v' = -b\rangle\, z > M}}$$

Deduction Modulo (Side Deduction)



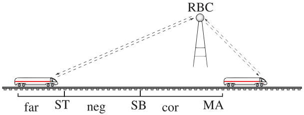
$$\dfrac{v \geq 0, z < M \to t \geq 0 \qquad \dfrac{v \geq 0, z < M \to -\frac{b}{2}t^2 + vt + z > M}{v \geq 0, z < M \to \langle z := -\frac{b}{2}t^2 + vt + z \rangle z > M}}{v \geq 0, z < M \to t \geq 0 \land \langle z := -\frac{b}{2}t^2 + vt + z \rangle z > M}$$

QE

$$\dfrac{\dfrac{\dfrac{v \geq 0, z < M \to \mathrm{QE}\left(\exists t \, (\ldots t \geq 0 \land -\frac{b}{2}t^2 + vt + z > M)\right)}{v \geq 0, z < M \to \exists t \geq 0 \, \langle z := -\frac{b}{2}t^2 + vt + z \rangle z > M}}{v \geq 0, z < M \to \langle z' = v, v' = -b \rangle z > M}}{v \geq 0 \land z < M \to \langle z' = v, v' = -b \rangle \, z > M}$$

start side

Deduction Modulo (Side Deduction)



$$\dfrac{v \geq 0, z < M \rightarrow {\color{red}t}{\geq}0 \qquad \dfrac{v \geq 0, z < M \rightarrow -\frac{b}{2}{\color{red}t}^2 + v{\color{red}t} + z > M}{v \geq 0, z < M \rightarrow \langle z := -\frac{b}{2}t^2 + vt + z\rangle z{>}M}}{v \geq 0, z < M \rightarrow t \geq 0 \land \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}$$

QE

start side

$$\dfrac{\dfrac{\dfrac{v \geq 0, z < M \rightarrow {\color{red}v^2 > 2b(M-z)}}{v \geq 0, z < M \rightarrow \exists t{\geq}0\, \langle z := -\frac{b}{2}t^2 + vt + z\rangle z > M}}{v \geq 0, z < M \rightarrow \langle z' = v, v' = -b\rangle z > M}}{v \geq 0 \land z < M \rightarrow \langle z' = v, v' = -b\rangle\ z > M}$$

Soundness

Theorem (Soundness)

d$\mathcal{L}$ *calculus is sound, i.e., all provable* d$\mathcal{L}$ *formulas are valid:*

$$\vdash \phi \textit{ implies } \vDash \phi$$

What about the converse?

Soundness

Theorem (Soundness)

d$\mathcal{L}$ *calculus is sound, i.e., all provable* d$\mathcal{L}$ *formulas are valid:*

$$\vdash \phi \textit{ implies } \vDash \phi$$

What about the converse?

$$(s := s + 2n + 1; n := n + 1)^* \quad \rightsquigarrow \quad s = n^2$$

Soundness

Theorem (Soundness)

$\mathrm{d}\mathcal{L}$ *calculus is sound, i.e., all provable* $\mathrm{d}\mathcal{L}$ *formulas are valid:*

$$\vdash \phi \text{ implies } \vDash \phi$$

What about the converse?

$$\begin{array}{lll} (s := s + 2n + 1; n := n + 1)^* & \leadsto & s = n^2 \\ x' = 5 & \leadsto & x(t) = 5t + x_0 \end{array}$$

Soundness

Theorem (Soundness)

d$\mathcal{L}$ *calculus is sound, i.e., all provable* d$\mathcal{L}$ *formulas are valid:*

$$\vdash \phi \text{ implies } \vDash \phi$$

What about the converse?

$$\begin{array}{lll} (s := s + 2n + 1; n := n + 1)^* & \leadsto & s = n^2 \\ x' = 5 & \leadsto & x(t) = 5t + x_0 \\ x' = x & \leadsto & x(t) = x_0 e^t \end{array}$$

Soundness

Theorem (Soundness)

d$\mathcal{L}$ *calculus is sound, i.e., all provable* d$\mathcal{L}$ *formulas are valid:*

$$\vdash \phi \text{ implies } \vDash \phi$$

What about the converse?

$$\begin{array}{lcl}
(s := s + 2n + 1; n := n + 1)^* & \leadsto & s = n^2 \\
x' = 5 & \leadsto & x(t) = 5t + x_0 \\
x' = x & \leadsto & x(t) = x_0 e^t \\
x'' = -x & \leadsto & x(t) = x_0 \cos t + x_0' \sin t
\end{array}$$

Soundness and Completeness

Theorem (Relative Completeness)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to differential equations.*

▸ Proof Outline 15p

Soundness and Completeness

Theorem (Relative Completeness)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to differential equations.*

Proof Outline 15p

Corollary (Proof-theoretical Alignment)

proving hybrid systems = proving dynamical systems!

Corollary (Compositionality)

hybrid systems can be verified by recursive decomposition

Discrete Completeness

Theorem (Relative Completeness / Continuous)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to differential equations.* ▸ Proof Outline

$$\vDash \phi \text{ iff } Taut_{FOD} \vdash \phi$$

Theorem (Relative Completeness / Discrete)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to discrete dynamics.* ▸ Proof Outline

$$\vDash \phi \text{ iff } Taut_{DL} \vdash \phi$$

Discrete Completeness

Theorem (Relative Completeness / Continuous)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to differential equations.* ▸ Proof Outline

$$\vDash \phi \text{ iff } Taut_{FOD} \vdash \phi$$

Theorem (Relative Completeness / Discrete)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to discrete dynamics.* ▸ Proof Outline

$$\vDash \phi \text{ iff } Taut_{DL} \vdash \phi$$

Corollary (Complete Proof-theoretical Alignment)

hybrid = continuous = discrete

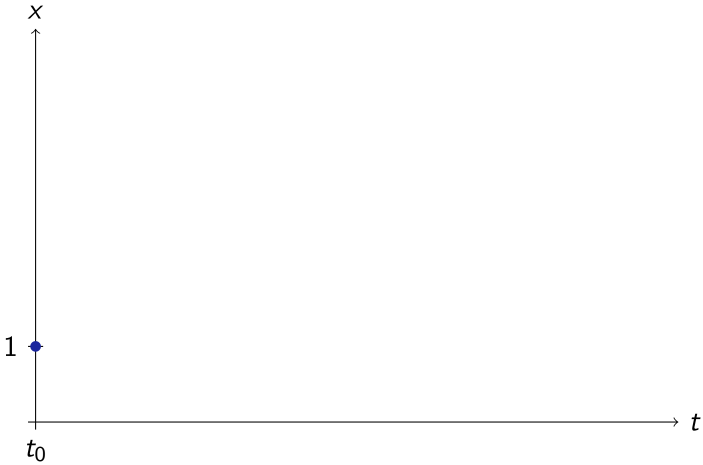
Corollary (Interdisciplinary Integrability)

"Discrete computer science + continuous control are integrable"

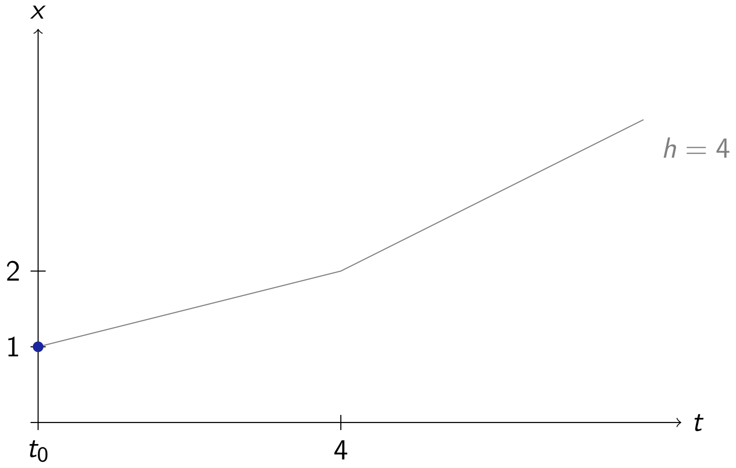
Proof of "hybrid = continuous = discrete"

Continuous Dynamics ≠ Discrete Dynamics

$$[x' = \frac{x}{4}]F$$

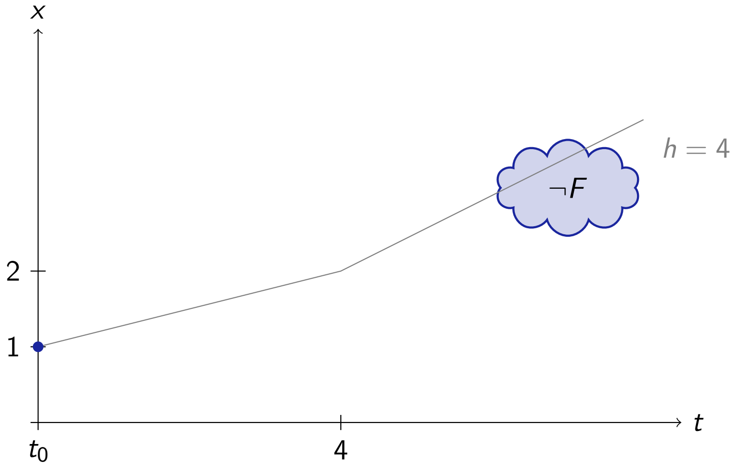


$$[x' = \frac{x}{4}]F \qquad [(x := x + h\frac{x}{4})^*]F$$



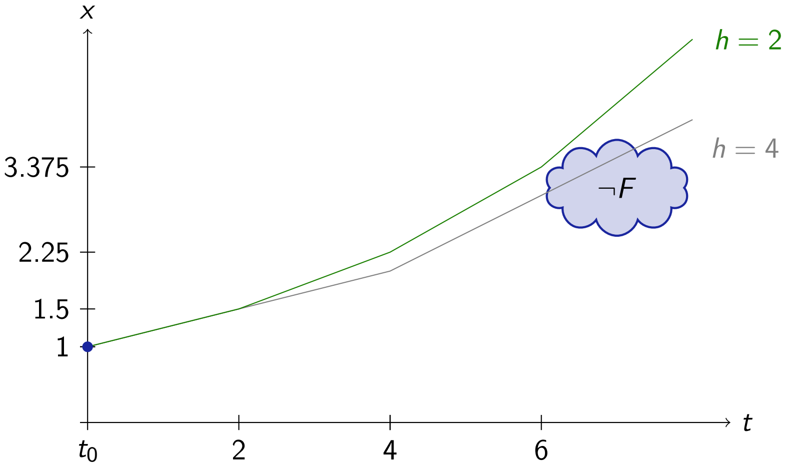
Continuous Dynamics $\neq$ Discrete Dynamics

$$[x' = \frac{x}{4}]F \quad \not\leftrightarrow \quad [(x := x + h\frac{x}{4})^*]F$$

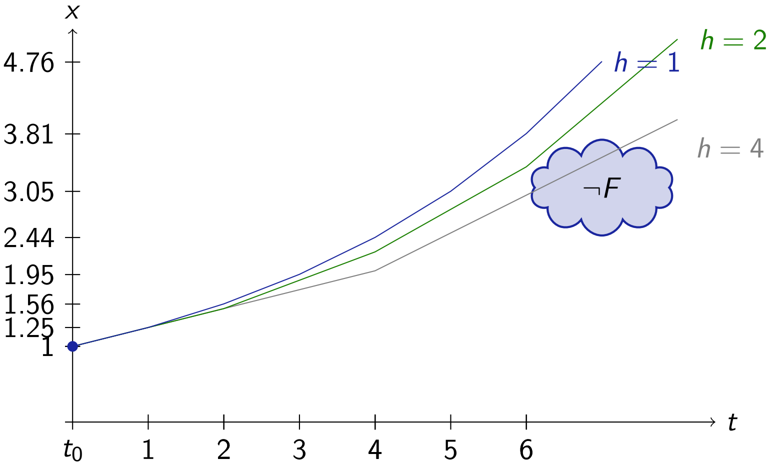


Continuous Dynamics $\neq$ Discrete Dynamics

$$[x' = \frac{x}{4}]F \qquad [(x := x + h\frac{x}{4})^*]F$$

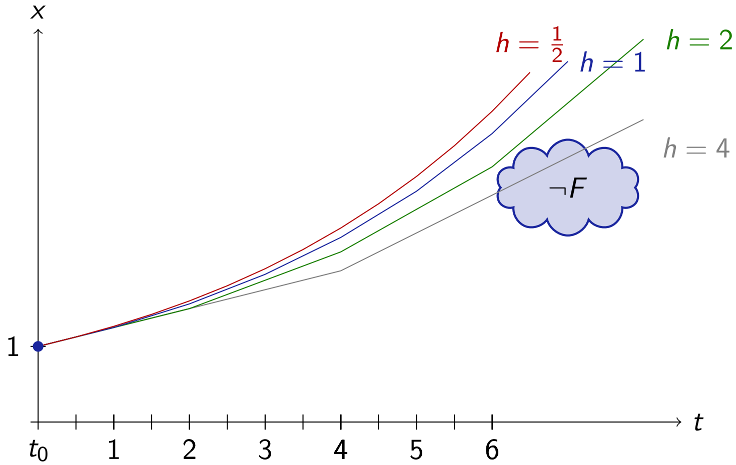


$$[x' = \frac{x}{4}]F \qquad [(x := x + h\frac{x}{4})^*]F$$



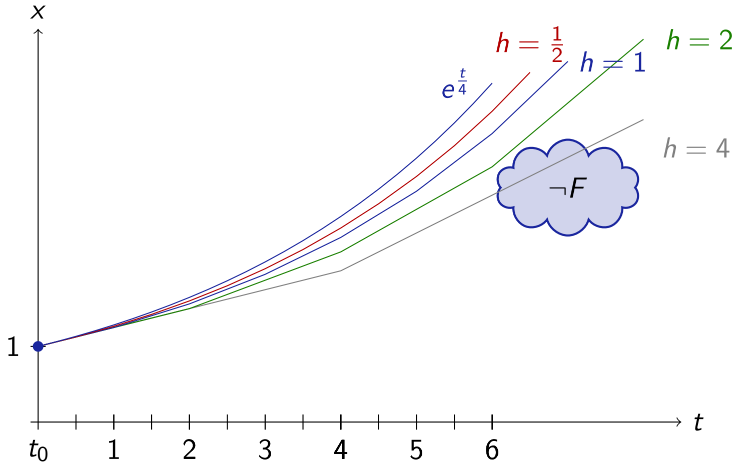
Continuous Dynamics $\neq$ Discrete Dynamics

$$[x' = \frac{x}{4}]F \qquad [(x := x + h\frac{x}{4})^*]F$$



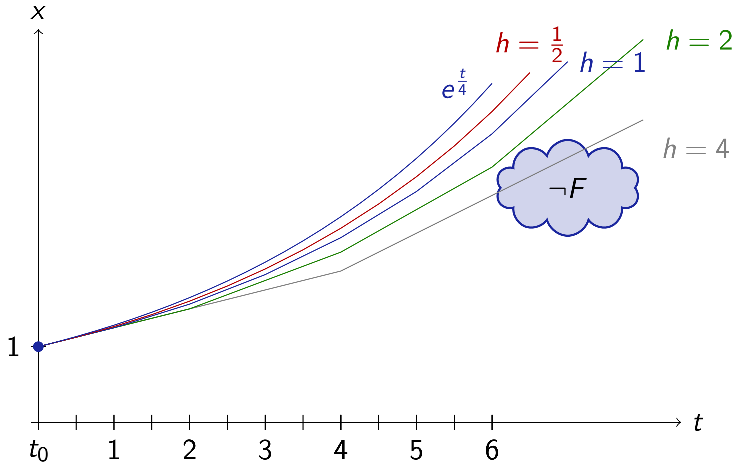
Continuous Dynamics $\neq$ Discrete Dynamics

$$[x' = \frac{x}{4}]F \quad \text{vs.} \quad [(x := x + h\frac{x}{4})^*]F$$

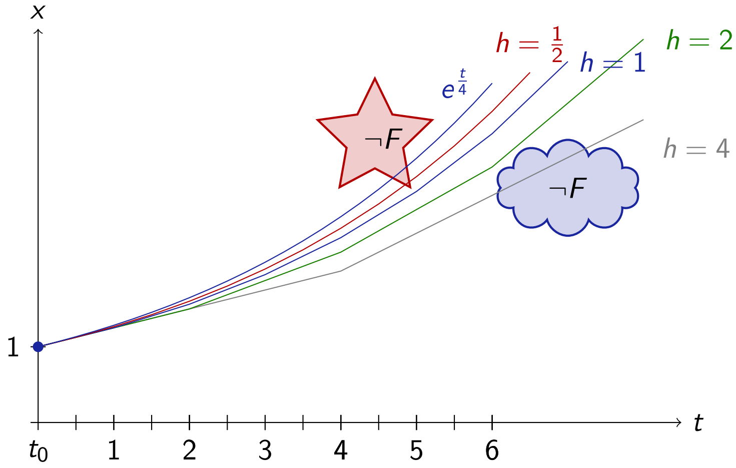


Continuous Dynamics $\neq$ Discrete Dynamics

$$[x' = \frac{x}{4}]F \quad \not\Rightarrow \quad [(x := x + h\frac{x}{4})^*]F$$



$$[x' = \frac{x}{4}]F \quad \not\leftarrow \quad [(x := x + h\frac{x}{4})^*]F$$



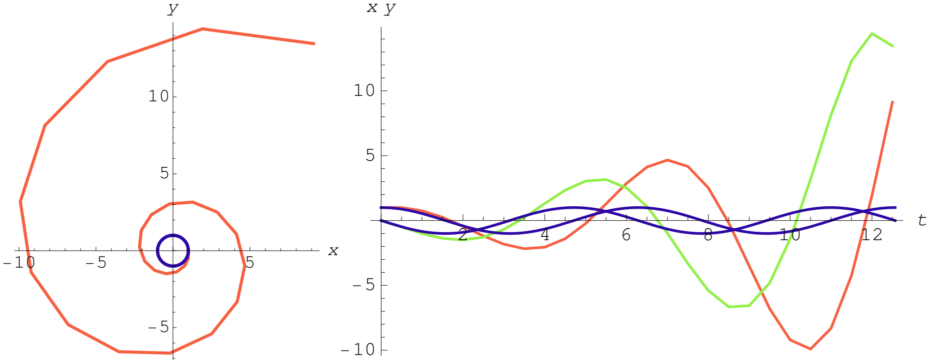
Discrete Euler Approximation Axiom $\overleftarrow{\Delta}$

$$\overleftarrow{\Delta} \quad [x' = f(x)]F \leftarrow \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*]F$$

$$\overleftarrow{\Delta} \quad [x' = f(x)]F \leftarrow \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*]F$$

Example (Insufficient, not global)

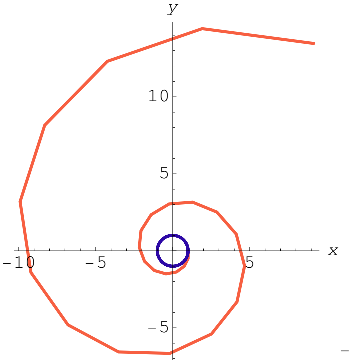
$$\vDash x^2 + y^2 \leq 1.1 \rightarrow [x' = y, y' = -x]x^2 + y^2 \leq 1.1$$

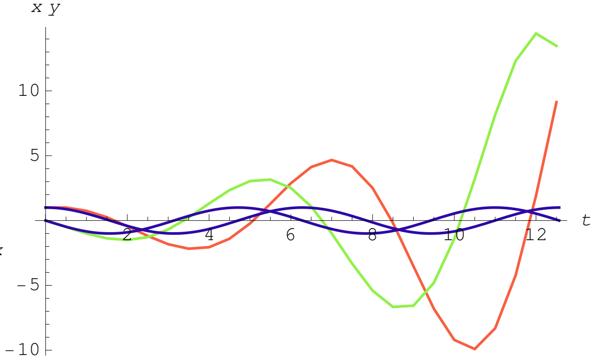


$$\overleftarrow{\Delta} \quad [x' = f(x)]F \leftarrow \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*]F \qquad \text{(closed)}$$

Example (Unsound for open F, only in closure)

$$\nvDash x = 1 \land y = 0 \rightarrow [x' = y, y' = -x](x \leq 0 \rightarrow x^2 + y^2 > 1)$$

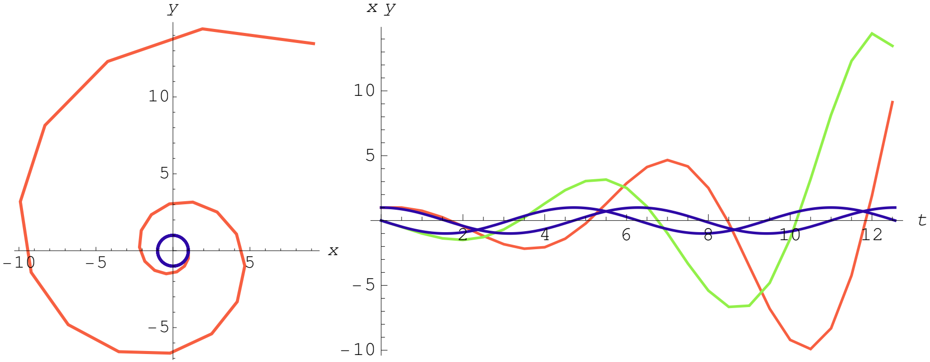




$$\overleftarrow{\Delta} \quad [x' = f(x)]F \leftarrow \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*]F \qquad \text{(closed)}$$

Example (Insufficient, not global)

$$\vDash x^2 + y^2 \leq 1.1 \rightarrow [x' = y, y' = -x]x^2 + y^2 \leq 1.1$$

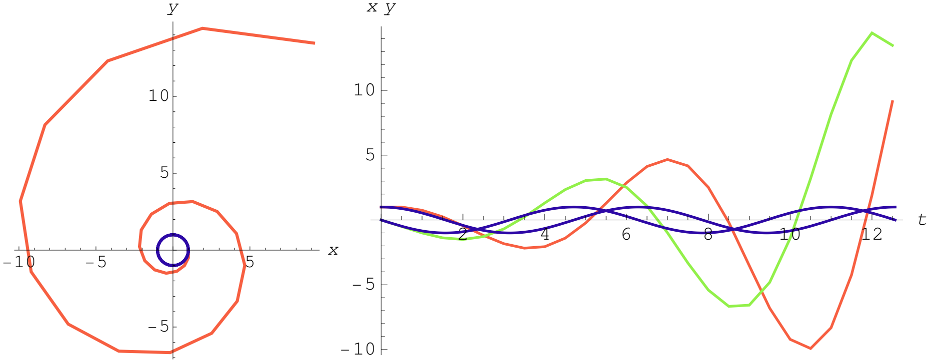


$$\overrightarrow{\Delta} \quad [x' = f(x)]F \\ \rightarrow \forall t{\geq}0\, \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*](t \geq 0 \rightarrow F)$$

$$\overrightarrow{\Delta} \quad [x' = f(x)]F \\ \rightarrow \forall t{\geq}0\, \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*](t \geq 0 \rightarrow F)$$

Example (Converse unsound for open F — $\overleftarrow{\Delta}$ for closed F)

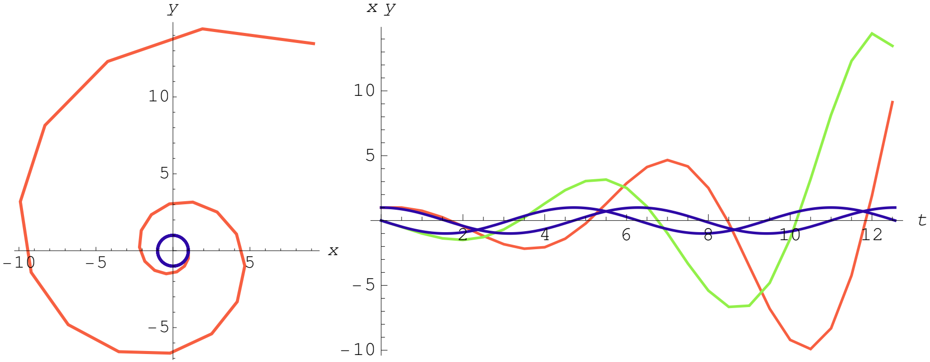
$$\nvDash x = 1 \land y = 0 \rightarrow [x' = y, y' = -x](x \leq 0 \rightarrow x^2 + y^2 > 1)$$



$\overrightarrow{\Delta}$ $[x' = f(x)]F$
$\rightarrow \forall t{\geq}0\, \exists h_0{>}0\, \forall 0{<}h{<}h_0\, [(x := x + hf(x))^*](t \geq 0 \rightarrow F)$ (open)

Example (Unsound for closed F, only holds in the limit)

$$\vDash x^2 + y^2 = 1 \rightarrow [x' = y, y' = -x]x^2 + y^2 = 1$$



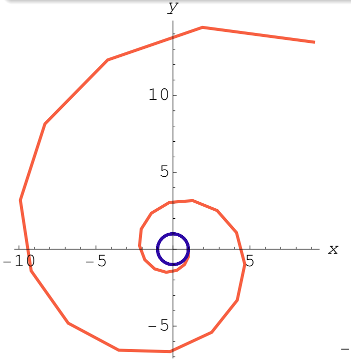
$$(\overleftrightarrow{\Delta}) \quad [x' = f(x)]F \\ \leftrightarrow \forall t{\geq}0 \exists \varepsilon{>}0 \exists h_0{>}0 \forall 0{<}h{<}h_0 [(x := x + hf(x))^*] \big(t{\geq}0 \rightarrow \neg\mathcal{U}_\varepsilon(\neg F)\big)$$

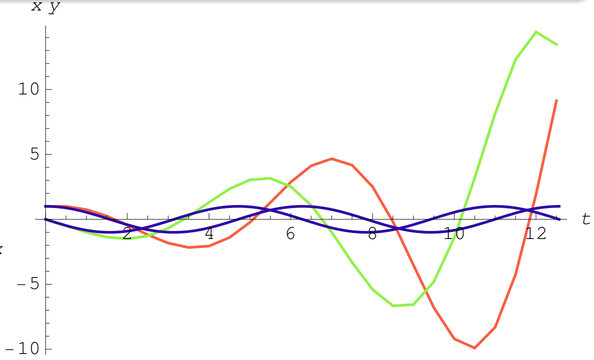
$$(\overleftrightarrow{\Delta}) \quad [x' = f(x)]F \\ \leftrightarrow \forall t \geq 0 \exists \varepsilon > 0 \exists h_0 > 0 \forall 0 < h < h_0 [(x := x + hf(x))^*](t \geq 0 \rightarrow \neg \mathcal{U}_\varepsilon(\neg F))$$

$$(\overleftrightarrow{\Delta}) \quad [x' = f(x)]F \leftrightarrow \forall t{\geq}0 \exists \varepsilon{>}0 \exists h_0{>}0 \forall 0{<}h{<}h_0 [(x := x + hf(x))^*](t{\geq}0 \rightarrow \neg\mathcal{U}_\varepsilon(\neg F))$$

Example (Time-uniform $\exists\varepsilon{>}0\forall t{\geq}0$ would be incomplete)

$$\vDash x^2 + y^2 < 1.1 \rightarrow [x' = y, y' = -x]x^2 + y^2 < 1.1$$

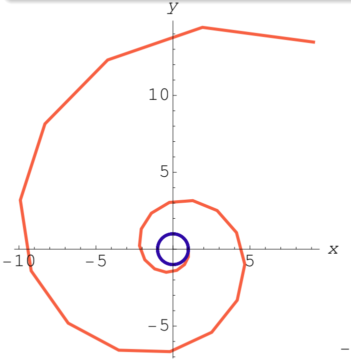


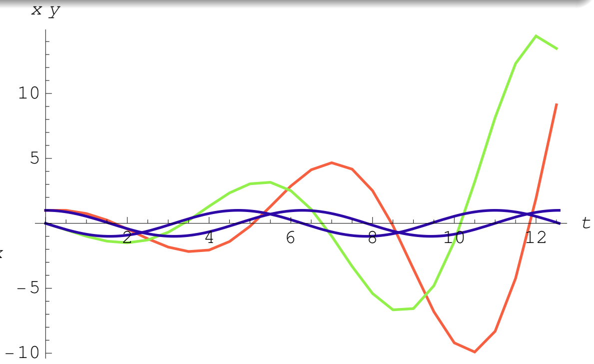


$$(\overleftrightarrow{\Delta}) \quad [x' = f(x)]F \qquad \textit{(open}$$
$$\leftrightarrow \forall t{\geq}0 \exists \varepsilon{>}0 \exists h_0{>}0 \forall 0{<}h{<}h_0 [(x := x + hf(x))^*](t{\geq}0 \rightarrow \neg\mathcal{U}_\varepsilon(\neg F))$$

Example (Insufficient for closed F)

$$\vDash x^2 + y^2 \leq 1 \rightarrow [x' = y, y' = -x] x^2 + y^2 \leq 1$$



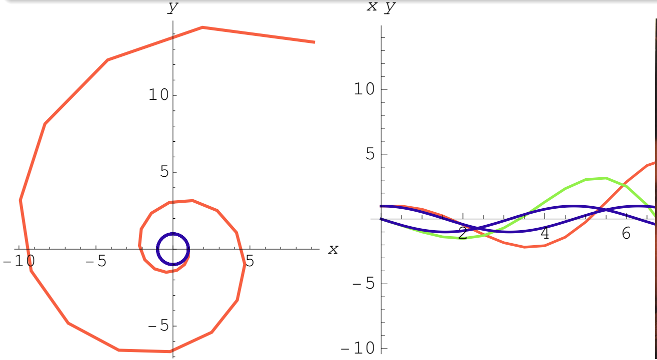


$(\overleftrightarrow{\Delta})$ $[x' = f(x)]F$ *(open*

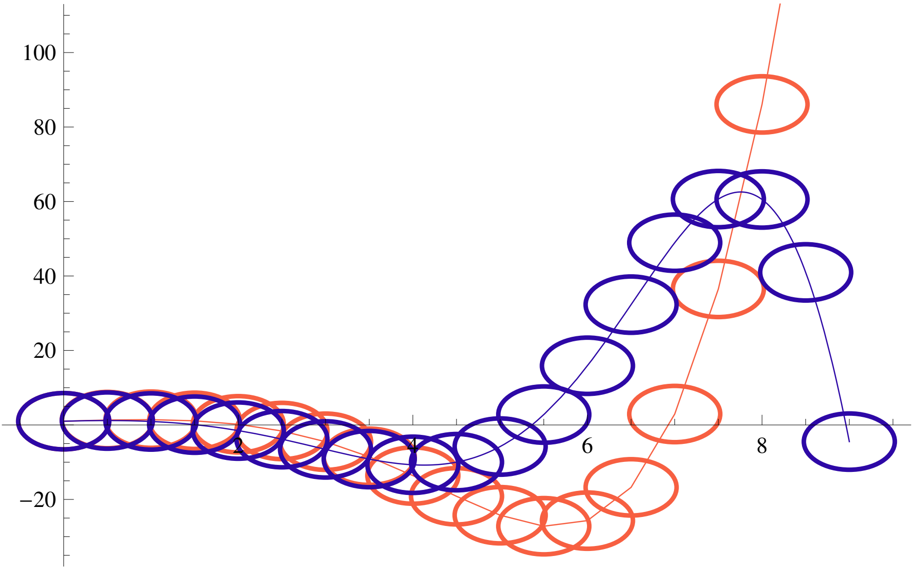
$\leftrightarrow \forall t{\geq}0 \exists \varepsilon{>}0 \exists h_0{>}0 \forall 0{<}h{<}h_0 [(x := x{+}hf(x))^*](t{\geq}0 \rightarrow \neg\mathcal{U}_\varepsilon(\neg F))$

Example (Insufficient for closed F)

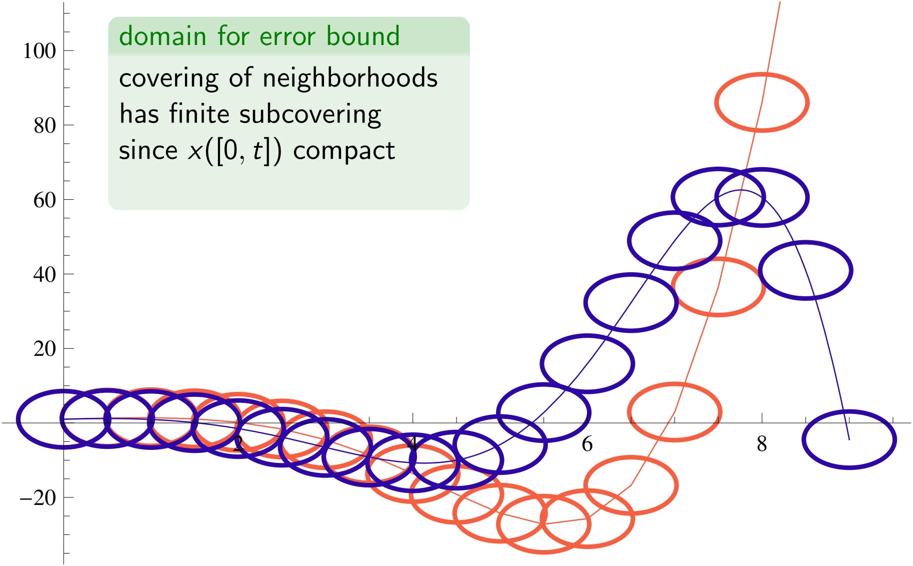
$$\vDash x^2 + y^2 \leq 1 \rightarrow [x' = y, y' = -x]x^2 + y^2 \leq 1$$



$\mathcal{R}$ $\overleftrightarrow{\Delta}$ Proof: Partial Covering for Solution, Approximation



$\mathcal{R}$ $\overleftrightarrow{\Delta}$ Proof: Partial Covering for Solution, Approximation



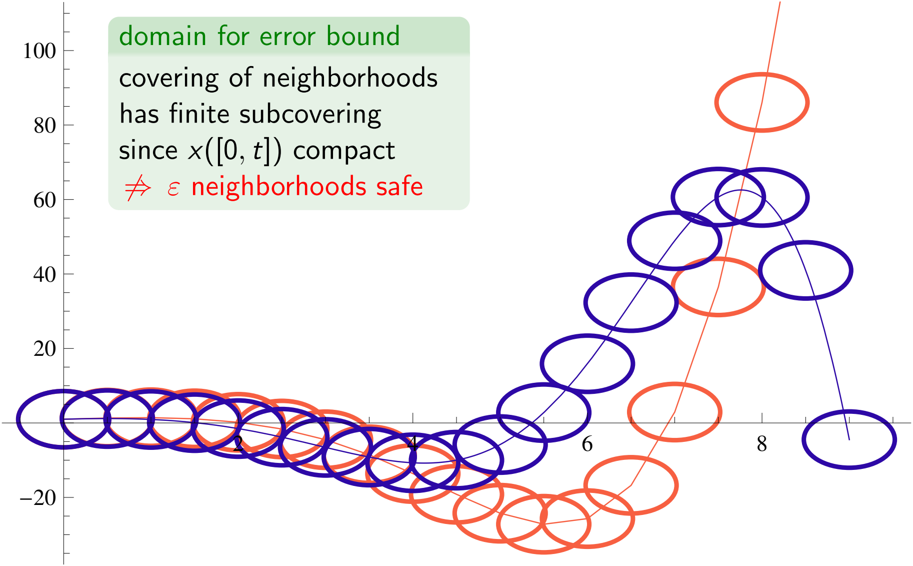
domain for error bound

covering of neighborhoods has finite subcovering since $x([0, t])$ compact

Proof: Partial Covering for Solution, Approximation

domain for error bound

covering of neighborhoods
has finite subcovering
since $x([0, t])$ compact
$\not\Rightarrow$ ε neighborhoods safe



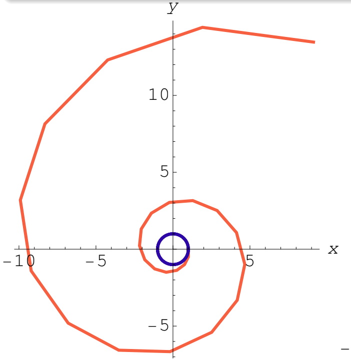
$\overleftrightarrow{\Delta}$ axiom for open F, but F may be closed

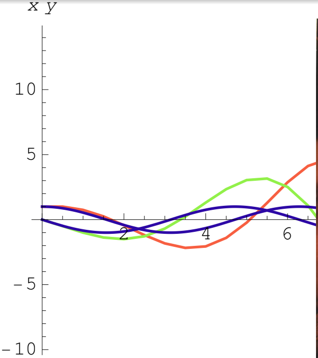
$(\overleftrightarrow{\Delta})$ $[x' = f(x)]F$ (open

$\leftrightarrow \forall t{\geq}0 \exists \varepsilon{>}0 \exists h_0{>}0 \forall 0{<}h{<}h_0 [(x := x{+}hf(x))^*](t{\geq}0 \to \neg \mathcal{U}_\varepsilon(\neg F))$

Example (Insufficient for closed F)

$$\vDash x^2 + y^2 \leq 1 \to [x' = y, y' = -x]x^2 + y^2 \leq 1$$





Closed Discrete Completeness (derivable)

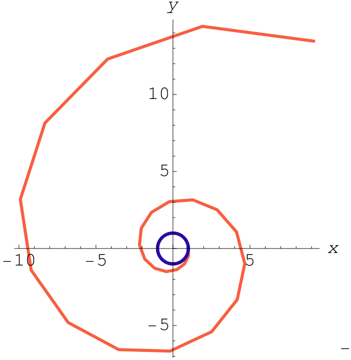
$$(\mathring{U}) \quad [x' = f(x)]F \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)]\mathcal{U}_{\check{\varepsilon}}(F) \qquad (\Leftarrow \text{B,V,G,K})$$

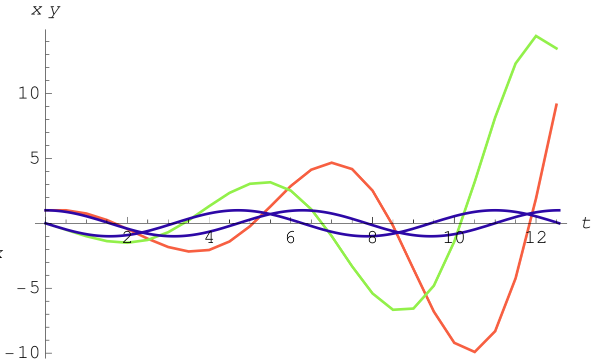
Closed Discrete Completeness (derivable)

$(\mathring{U})\quad [x' = f(x)]F \leftrightarrow \forall \check{\varepsilon}>0\, [x' = f(x)]\mathcal{U}_{\check{\varepsilon}}(F)$ $\qquad (\Leftarrow \text{B,V,G,K})$

Example (Closed $\rightsquigarrow$ Quantified Open)

$$\vDash x^2 + y^2 \leq 1 \rightarrow [x' = y, y' = -x]x^2 + y^2 \leq 1$$



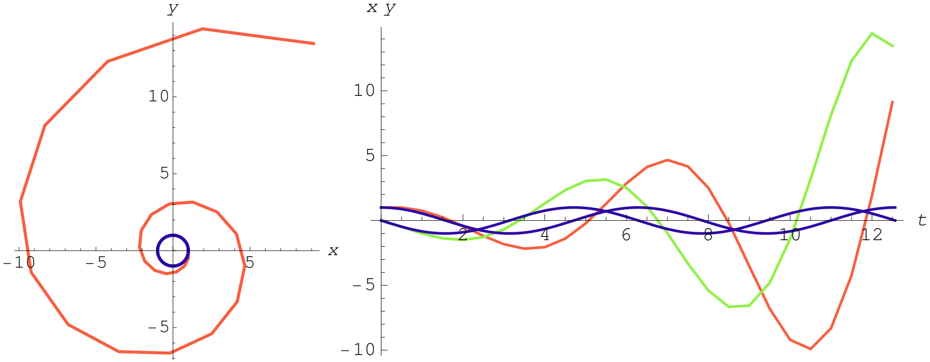


Closed Discrete Completeness (derivable)

$(\mathring{U}) \quad [x' = f(x)]F \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)]\mathcal{U}_{\check{\varepsilon}}(F) \qquad (\Leftarrow \text{B,V,G,K})$

Example (Closed $\rightsquigarrow$ Quantified Open)

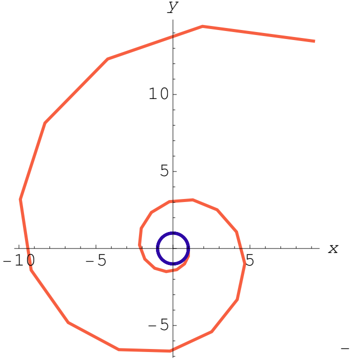
$$\vDash x^2 + y^2 \leq 1 \rightarrow [x' = y, y' = -x]\forall \check{\varepsilon}{>}0\, x^2 + y^2 < 1 + \check{\varepsilon}$$

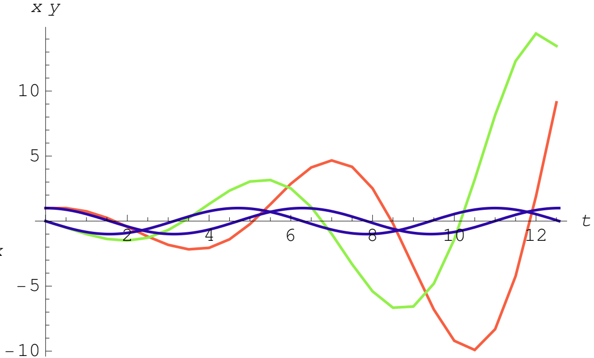


$(\mathring{U})$ $[x' = f(x)]F \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)]\mathcal{U}_{\check{\varepsilon}}(F)$ $(\Leftarrow$ B,V,G,K)

Example (Closed $\rightsquigarrow$ Quantified Open)

$$\vDash x^2 + y^2 \leq 1 \rightarrow \forall \check{\varepsilon}{>}0\, [x' = y, y' = -x]x^2 + y^2 < 1 + \check{\varepsilon}$$





$\overleftrightarrow{\Delta}$ axiom for open/closed F, but otherwise?

Locally Closed Discrete Completeness (derivable)

Example (Locally Closed $\leadsto$ Open, Closed)

$$\vDash O \land C \to [x' = y, y' = -x](O \land C)$$

$$([]\wedge) \quad [\alpha](O \wedge C) \leftrightarrow [\alpha]O \wedge [\alpha]C \qquad (\Leftarrow \text{K})$$

Example (Locally Closed $\leadsto$ Open, Closed)

$$\vDash O \wedge C \rightarrow [x' = y, y' = -x](O \wedge C)$$

Locally Closed Discrete Completeness (derivable)

$$([]\wedge) \quad [\alpha](O \wedge C) \leftrightarrow [\alpha]O \wedge [\alpha]C \qquad (\Leftarrow \text{K})$$

Example (Locally Closed $\rightsquigarrow$ Open, Closed)

$$\vDash O \wedge C \rightarrow [x' = y, y' = -x]O \wedge [x' = y, y' = -x]C$$

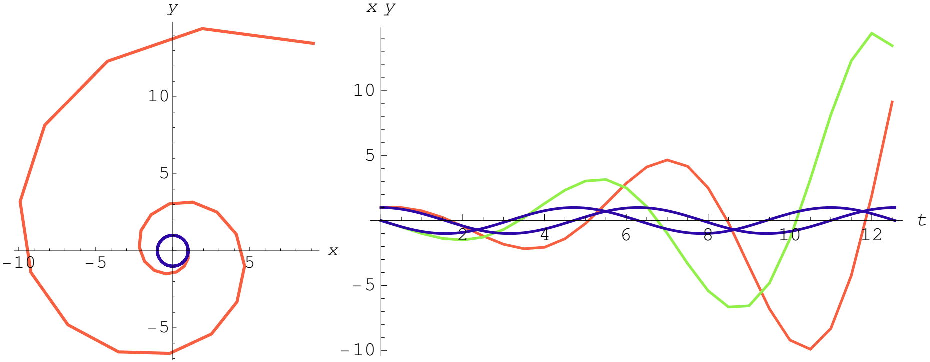
Semialgebraic Discrete Completeness (derivable)

$$(\check{U}) \quad [x' = f(x)](O \lor C) \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)](O \lor \mathcal{U}_{\check{\varepsilon}}(C)) \qquad (\Leftarrow \text{ B,V,G,K})$$

$$(\check{U}) \quad [x' = f(x)](O \vee C) \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)](O \vee \mathcal{U}_{\check{\varepsilon}}(C)) \qquad (\Leftarrow \text{ B,V,G,K})$$

Example ((Open $\vee$ Closed) $\rightsquigarrow$ Quantified Open)

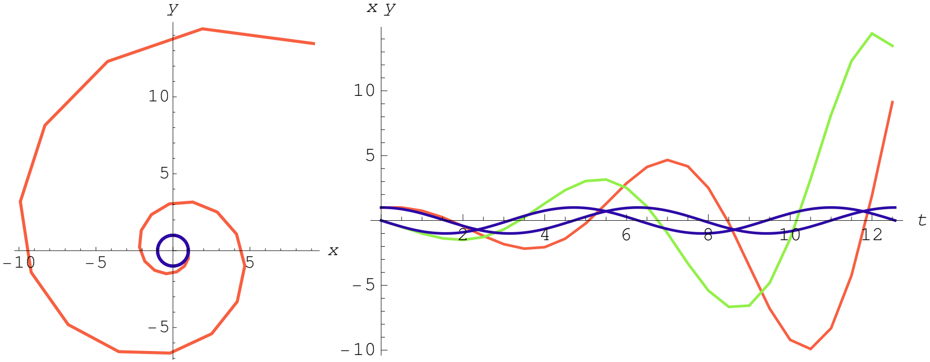
$$\vDash O \vee C \rightarrow [x' = y, y' = -x](O \vee C)$$



$$(\check{U}) \quad [x' = f(x)](O \lor C) \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)](O \lor \mathcal{U}_{\check{\varepsilon}}(C)) \qquad (\Leftarrow \text{ B,V,G,K})$$

Example ((Open ∨ Closed) ⇝ Quantified Open)

$$\vDash O \lor C \to [x' = y, y' = -x](O \lor \forall \check{\varepsilon}{>}0\, \mathcal{U}_{\check{\varepsilon}}(C))$$

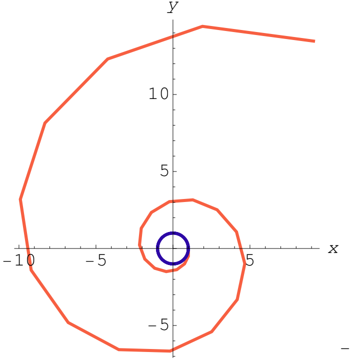


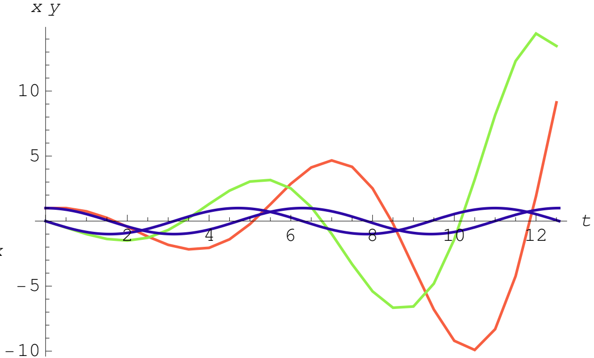
Semialgebraic Discrete Completeness (derivable)

$$(\check{U}) \quad [x' = f(x)](O \vee C) \leftrightarrow \forall \check{\varepsilon}{>}0\, [x' = f(x)](O \vee \mathcal{U}_{\check{\varepsilon}}(C)) \qquad (\Leftarrow \text{ B,V,G,K})$$

Example ((Open ∨ Closed) ⇝ Quantified Open)

$$\vDash O \vee C \rightarrow [x' = y, y' = -x]\forall \check{\varepsilon}{>}0\, (O \vee \mathcal{U}_{\check{\varepsilon}}(C))$$



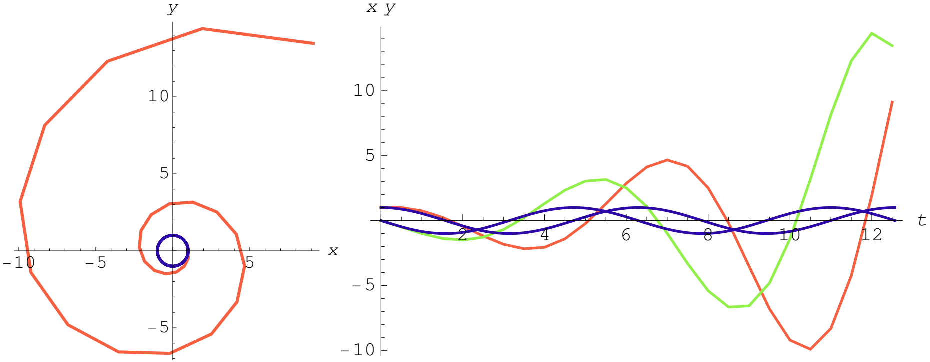


Semialgebraic Discrete Completeness (derivable)

$$(\check{U})\quad [x' = f(x)](O \lor C) \leftrightarrow \forall \check{\varepsilon}{>}0\,[x' = f(x)](O \lor \mathcal{U}_{\check{\varepsilon}}(C)) \qquad (\Leftarrow\ \text{B,V,G,K})$$

Example ((Open ∨ Closed) ⇝ Quantified Open)

$$\vDash O \lor C \to \forall \check{\varepsilon}{>}0\,[x' = y, y' = -x](O \lor \mathcal{U}_{\check{\varepsilon}}(C))$$



$\overleftrightarrow{\Delta}$ axiom for semialgebraic F, but otherwise?

Discrete Completeness

Theorem (Relative Completeness / Continuous)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to* *differential equations.*

▸ Proof Outline 6p

$$\vDash \phi \textit{ implies } Taut_{FOD} \vdash \phi$$

Theorem (Relative Completeness / Discrete)

d$\mathcal{L}$ *calculus* $+ \overleftrightarrow{\Delta}$ *is a sound & complete axiomatization of hybrid systems relative to* *discrete dynamics.*

▸ Proof Outline +5p

$$\vDash \phi \textit{ implies } Taut_{DL} \vdash \phi$$

Proof Sketch.

Talked about 0-order semialgebraic

Paper proves $\forall, \exists$ ...

Paper proves $[\alpha], \langle\alpha\rangle$ with hybrid system α ...

Paper proves nesting ... □

Discrete Completeness

Theorem (Relative Completeness / Continuous)

d$\mathcal{L}$ *calculus is a sound & complete axiomatization of hybrid systems relative to differential equations.* ▸ Proof Outline 6p

$$\vDash \phi \text{ implies } Taut_{FOD} \vdash \phi$$

Theorem (Relative Completeness / Discrete)

d$\mathcal{L}$ *calculus* $+ \overleftrightarrow{\Delta}$ *is a sound & comple*

relative to discrete dynamics.

$\vDash \phi$ *implies*

Proof Sketch.

Talked about 0-order semialgebraic

Paper proves $\forall, \exists$ ...

Paper proves $[\alpha], \langle\alpha\rangle$ with hybrid syst

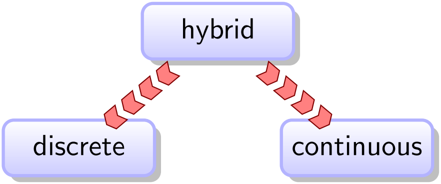
Paper proves nesting ...

Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD\ \ \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL\ \ \vDash \phi \leftrightarrow \phi^\#$$

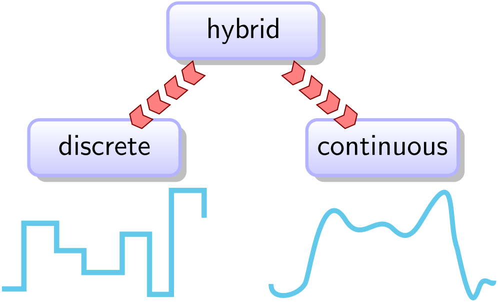


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD \ \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \ \vDash \phi \leftrightarrow \phi^\#$$

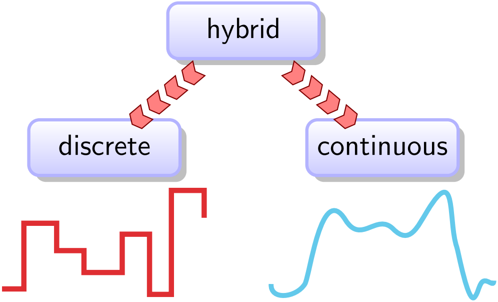


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD \quad \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \quad \vDash \phi \leftrightarrow \phi^\#$$

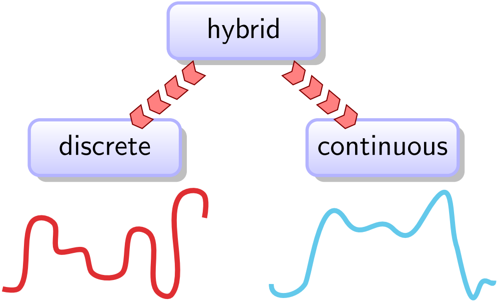


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD \quad \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \quad \vDash \phi \leftrightarrow \phi^\#$$

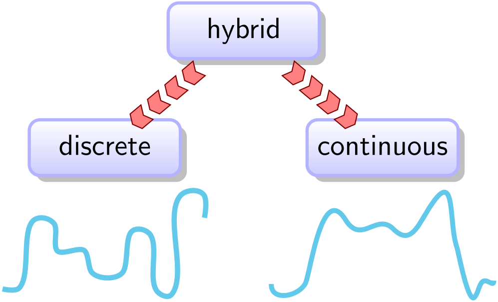


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD \quad \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \quad \vDash \phi \leftrightarrow \phi^\#$$

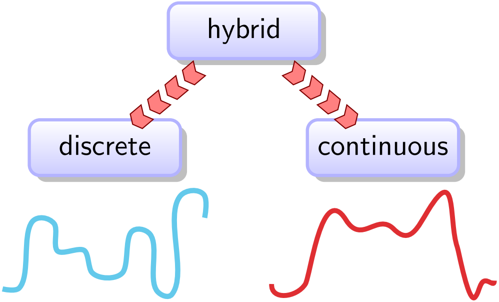


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ (constructively) expressible in FOD and in DL:

$$\forall\phi\ \exists\phi^\flat \in FOD \quad \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \quad \vDash \phi \leftrightarrow \phi^\#$$

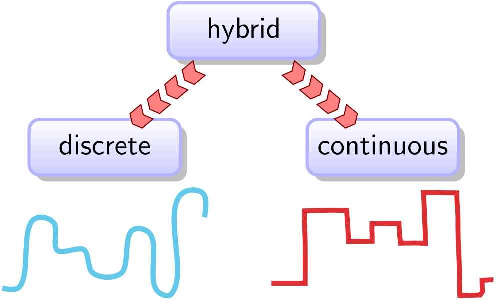


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ *(constructively) expressible in FOD and in DL:*

$$\forall\phi\ \exists\phi^\flat \in FOD \ \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in DL \ \vDash \phi \leftrightarrow \phi^\#$$

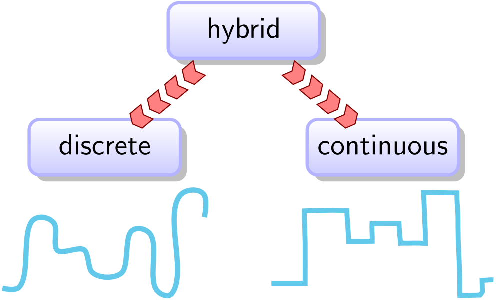


Equi-expressibility

Theorem (Equi-expressibility)

d$\mathcal{L}$ (constructively) expressible in FOD and in DL:

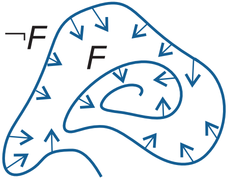
$$\forall\phi\ \exists\phi^\flat \in \mathit{FOD}\ \ \vDash \phi \leftrightarrow \phi^\flat$$
$$\forall\phi\ \exists\phi^\# \in \mathit{DL}\ \ \vDash \phi \leftrightarrow \phi^\#$$

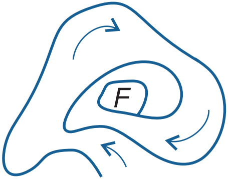


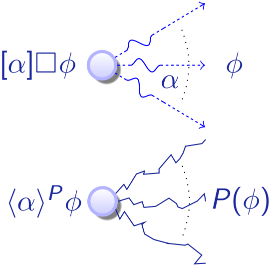
Relative Decidability

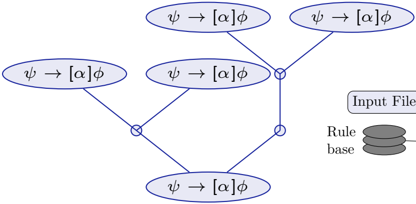
Theorem (Relative Decidability)

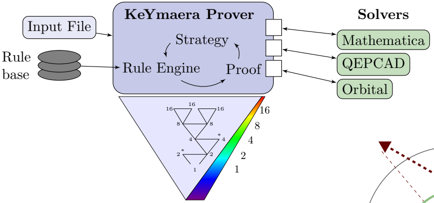
Validity of d$\mathcal{L}$ *sentences is decidable relative to FOD or DL.*

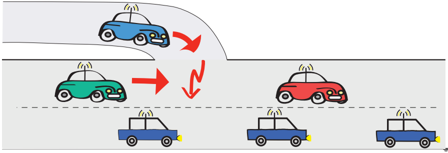

¬F
F

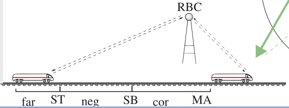

F

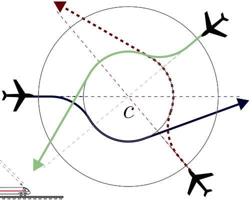

$[\alpha]\square\phi$
α
ϕ
$\langle\alpha\rangle^P\phi$
$P(\phi)$


$\psi \to [\alpha]\phi$
$\psi \to [\alpha]\phi$
$\psi \to [\alpha]\phi$
$\psi \to [\alpha]\phi$
$\psi \to [\alpha]\phi$

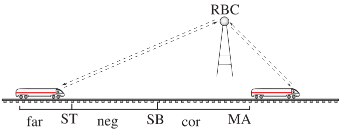

KeYmaera Prover
Input File
Rule base
Strategy
Rule Engine
Proof
Solvers
Mathematica
QEPCAD
Orbital
16
8
4
2
1

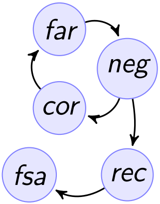


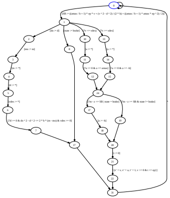

RBC
far
ST
neg
SB
cor
MA

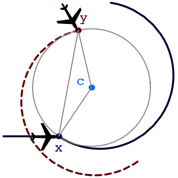

c

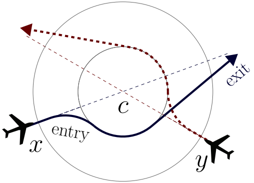
Successful Hybrid Systems Proofs

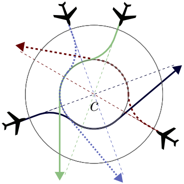


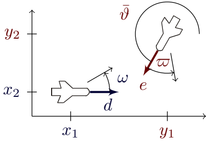


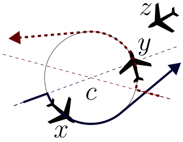


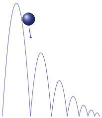


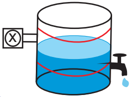




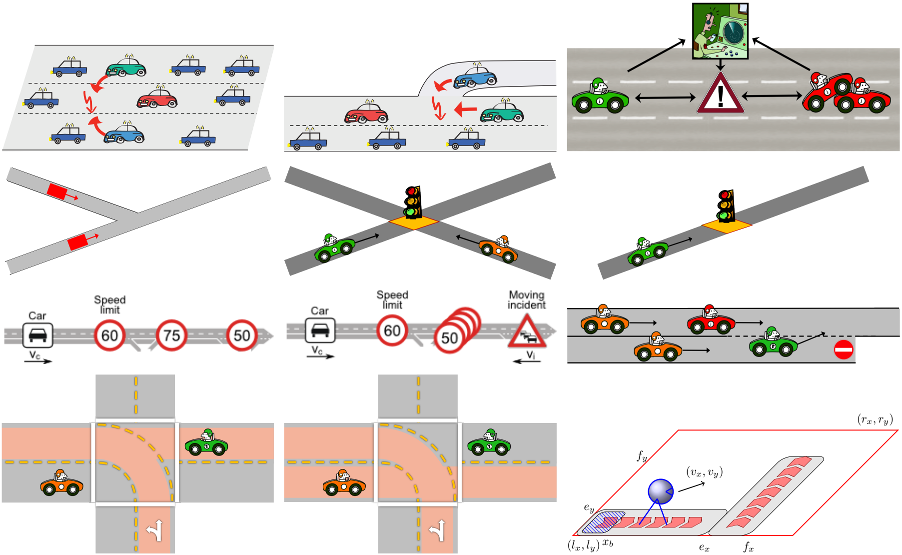








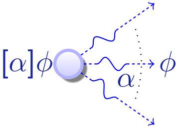
Successful Hybrid Systems Proofs



Logical Analysis of Hybrid Systems

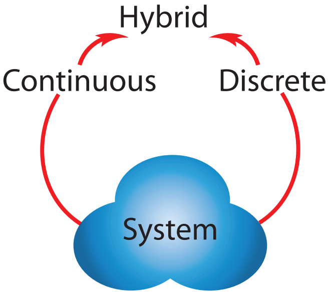
differential dynamic logic

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



proof-theoretical alignment

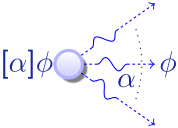
hybrid = continuous = discrete



Logical Analysis of Hybrid Systems

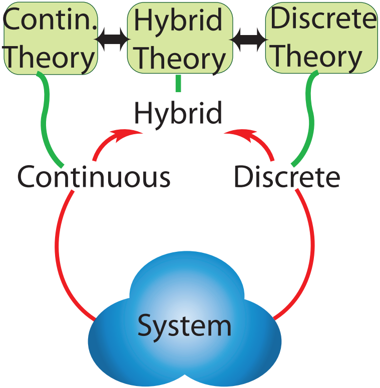
differential dynamic logic

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



proof-theoretical alignment

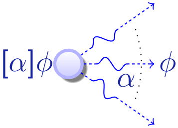
hybrid = continuous = discrete



Logical Analysis of Hybrid Systems

differential dynamic logic

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

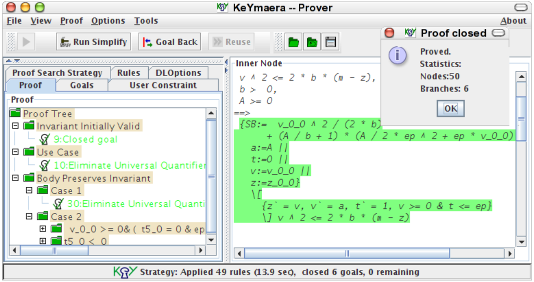


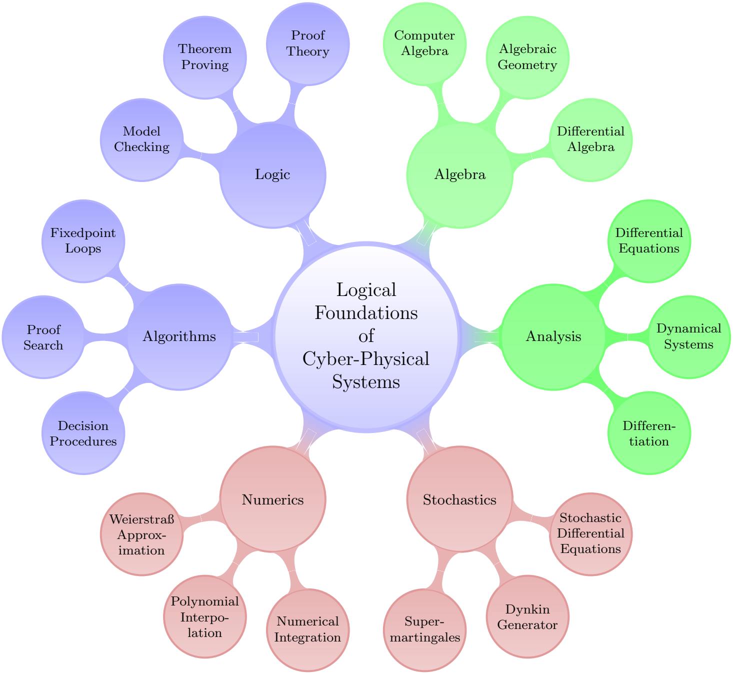
proof-theoretical alignment

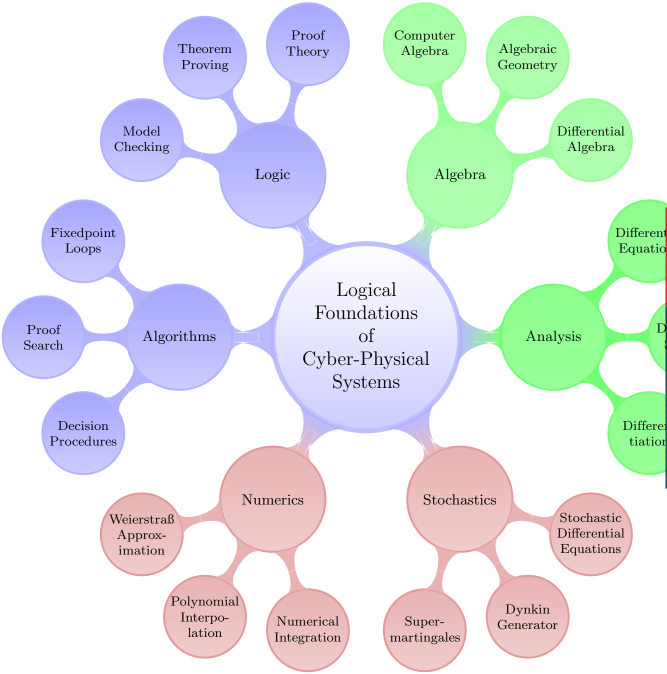
hybrid = continuous = discrete

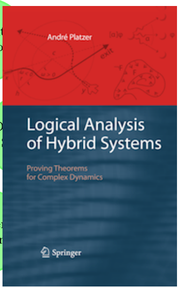
- Safety-critical systems
- Proof to be sure
- Proof to find bugs
- Proof to find constraints
- Logic for hybrid systems++
- Compositional proofs

KeYmaera




Logical Foundations of Cyber-Physical Systems
Logic
Model Checking
Theorem Proving
Proof Theory
Algebra
Computer Algebra
Algebraic Geometry
Differential Algebra
Analysis
Differential Equations
Dynamical Systems
Differen- tiation
Stochastics
Stochastic Differential Equations
Dynkin Generator
Super- martingales
Numerics
Weierstraß Approx- imation
Polynomial Interpo- lation
Numerical Integration
Algorithms
Fixedpoint Loops
Proof Search
Decision Procedures


Logical Foundations of Cyber-Physical Systems
Logic
Proof Theory
Theorem Proving
Model Checking
Algebra
Computer Algebra
Algebraic Geometry
Differential Algebra
Analysis
Algorithms
Fixedpoint Loops
Proof Search
Decision Procedures
Numerics
Weierstraß Approx-imation
Polynomial Interpo-lation
Numerical Integration
Stochastics
Stochastic Differential Equations
Dynkin Generator
Super-martingales


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

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

André Platzer.
Logics of dynamical systems.
In LICS [1], pages 13–24.

André Platzer.
The complete proof theory of hybrid systems.
In LICS [1], pages 541–550.

André Platzer.
Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics.
Springer, Heidelberg, 2010.

André Platzer.
Differential dynamic logic for hybrid systems.
J. Autom. Reas., 41(2):143–189, 2008.

André Platzer.
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 20(1):309–352, 2010.
Advance Access published on November 18, 2008.

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

André Platzer and Jan-David Quesel.
KeYmaera: A hybrid theorem prover for hybrid systems.
In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, *IJCAR*, volume 5195 of *LNCS*, pages 171–178. Springer, 2008.

André Platzer.
Differential dynamic logic for verifying parametric hybrid systems.
In Nicola Olivetti, editor, *TABLEAUX*, volume 4548 of *LNCS*, pages 216–232. Springer, 2007.

André Platzer.

Quantified differential dynamic logic for distributed hybrid systems.
In Anuj Dawar and Helmut Veith, editors, *CSL*, volume 6247 of *LNCS*, pages 469–483. Springer, 2010.

André Platzer.
A complete axiomatization of quantified differential dynamic logic for distributed hybrid systems.
Logical Methods in Computer Science, 2012.
Special issue for selected papers from CSL'10.

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.

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

André Platzer.
Logics of dynamical systems.

In LICS [1], pages 13–24.

André Platzer.
The complete proof theory of hybrid systems.
In LICS [1], pages 541–550.

André Platzer.
Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics.
Springer, Heidelberg, 2010.

André Platzer.
Differential dynamic logic for hybrid systems.
J. Autom. Reas., 41(2):143–189, 2008.

André Platzer.
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 20(1):309–352, 2010.
Advance Access published on November 18, 2008.

André Platzer and Edmund M. Clarke.
Computing differential invariants of hybrid systems as fixedpoints.

Form. Methods Syst. Des., 35(1):98–120, 2009.
Special issue for selected papers from CAV'08.

André Platzer and Jan-David Quesel.
KeYmaera: A hybrid theorem prover for hybrid systems.
In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, *IJCAR*, volume 5195 of *LNCS*, pages 171–178. Springer, 2008.

André Platzer.
Differential dynamic logic for verifying parametric hybrid systems.
In Nicola Olivetti, editor, *TABLEAUX*, volume 4548 of *LNCS*, pages 216–232. Springer, 2007.

André Platzer.
Quantified differential dynamic logic for distributed hybrid systems.
In Anuj Dawar and Helmut Veith, editors, *CSL*, volume 6247 of *LNCS*, pages 469–483. Springer, 2010.

André Platzer.
A complete axiomatization of quantified differential dynamic logic for distributed hybrid systems.

Logical Methods in Computer Science, 2012.
Special issue for selected papers from CSL'10.

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.

Outline Background

Outline Background

	Op	Par	T	Cl	Tec	Aut	Cex	Dim	
HenzingerH94, HyTech	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$		LHA
LafferrierePY99	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$		$\checkmark$		forgetful reset
Fränzle99	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$		$\checkmark$	$\times$	robust systems
CKrogh03, CheckMate	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$		polyhedral
Frehse05, PHAVer	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$	8	LHA (+affine)
MysorePM05	$\checkmark$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\bullet$	$\checkmark$	4	bounded prefix
TomlinPS98,MBT05	$\circ$	$\times$	$\times$	$\times$	$\circ$	$\circ$	$\bullet$	4	HJB numPDE
RatschanS07, HSolver	$\checkmark$	$\times$		$\times$	$\checkmark$	$\checkmark$	$\times$	4	interval
MannaS98, STeP	$\checkmark$			$\times$	$\checkmark$	$\circ$	$\times$	7	inv$\mapsto$VCG, flat
ÁbrahámSH01, PVS	$\bullet$			$\times$	$\bullet$	$\circ$	$\times$	$\approx$9	HA$\hookrightarrow$PVS, -“-
ZhouRH92, EDC	$\times$	$\bullet$	$\checkmark$	..	$\times$	$\times$	$\times$	$\times$	no maths
DavorenN00, Lμ	$\times$	$\times$		$\checkmark$	$\circ$	$\times$	$\times$	$\times$	prop. H-semantics
RönkköRS03, HGC	$\checkmark$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	HGC$\hookrightarrow$HOL
SSManna04	$\bullet$	$\circ$		$\times$	$\checkmark$		$\times$	4/1	equational system
CTiwari05	$\bullet$	$\circ$		$\times$	$\checkmark$		$\times$	6/0	linear, -”-
PrajnaJP07, barrier	$\bullet$	$\times$		$\times$	$\bullet$		$\times$	3	needs 10000-dim
d$\mathcal{L}$ & dTL	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\bullet$	$\times$	28	expr., compos.

	Dom	Op	Base	Modal	Quant	Cmpl	Aut
DL	$\mathbb{N}$		$FOL_{(\mathbb{N})}$		FV+unify	$/\mathbb{N}$	
$d\mathcal{L}$	$\mathbb{R}$	x'	$FOL_{\mathbb{R}}$	ODE	FV+requant+QE	/ODE	IBC

Outline Background

Soundness

Proof (Soundness).

- $x' = f(x)$
- Side deductions
- Free variables & Skolemisation

□

Return

Incompleteness

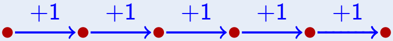
Theorem

Discrete fragment and continuous fragment of $\mathrm{d}\mathcal{L}$ *characterize* $\mathbb{N}$

Proof.

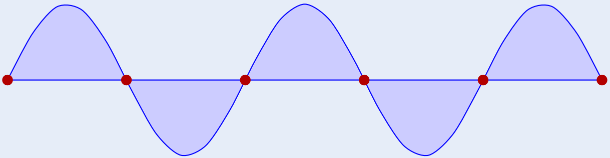
Discrete fragment:

$$\langle (x := x+1)^* \rangle\ x = n$$



Continuous fragment:

$$\langle s'' = -s, \tau' = 1 \rangle (s = 0 \land \tau = n) \qquad \leadsto s = \sin$$



Outline Background

Incomplete! But are we missing proof rules?

Incomplete! But are we missing proof rules?

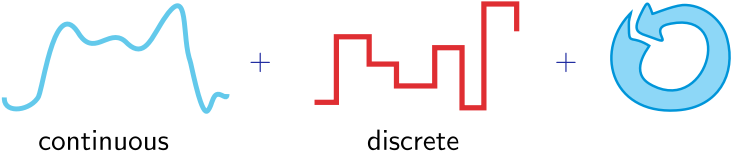
Relativity

Cook,Harel: discrete-DL/$\text{data}_{\mathbb{N}}$ hybrid-d$\mathcal{L}$/$\text{data}_{\mathbb{R}}$??

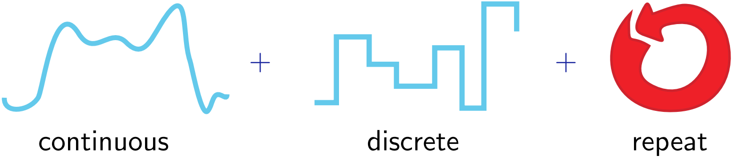
Sources of Incompleteness

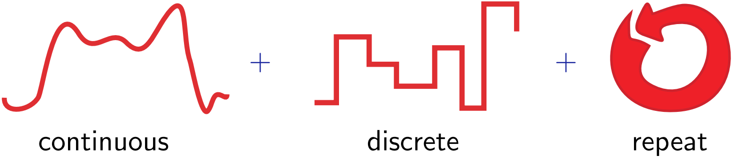
continuous

Sources of Incompleteness

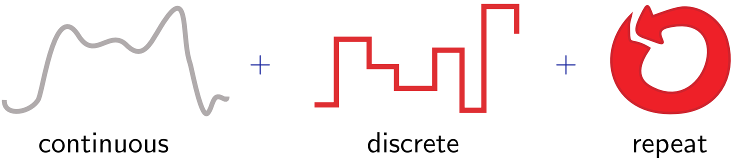


Sources of Incompleteness




continuous
+
discrete
+
repeat

Relative Completeness



Relative Completeness

continuous discrete repeat

Theorem (Relative Completeness)

d$\mathcal{L}$ calculus is complete relative to first-order logic of differential equations.

$$\vDash \phi \quad \text{iff} \quad \mathit{Taut}_{\mathrm{FOD}} \vdash \phi$$

where $\mathrm{FOD} = \mathrm{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

▸ Proof Outline 15p

continuous   discrete   repeat

$\Downarrow$

Theorem (Relative Completeness)

d$\mathcal{L}$ calculus is complete relative to first-order logic of differential equations.

$$\vDash \phi \quad \text{iff} \quad Taut_{\mathrm{FOD}} \vdash \phi$$

where $\quad \mathrm{FOD} = \mathrm{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

▸ Proof Outline 15p

$\Downarrow$

Relativity

Cook,Harel: discrete-DL/data P.: hybrid-d$\mathcal{L}$/differential equations

Relative Completeness

Theorem (Relative Completeness)

d$\mathcal{L}$ calculus is complete relative to first-order logic of differential equations.

$$\vDash \phi \quad \text{iff} \quad Taut_{\text{FOD}} \vdash \phi$$

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

▸ Proof Outline 15p

$$\Downarrow$$

Corollary (Proof-theoretical Alignment)

verification of hybrid systems = verification of dynamical systems!

Theorem (Relative Completeness)

d$\mathcal{L}$ calculus is complete relative to first-order logic of differential equations.

$$\vDash \phi \quad \text{iff} \quad Taut_{\mathrm{FOD}} \vdash \phi$$

where $\mathrm{FOD} = \mathrm{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

Proof Outline 15p

$\Downarrow$

Corollary (Deductive Power)

d$\mathcal{L}$ calculus is *supremal hybrid* verification technique

Relative Completeness Proof

$$\vDash \phi \quad \text{iff} \quad Taut_{\text{FOD}} \vdash \phi$$

$$\text{where} \quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$$

Proof (Relative Completeness, 10 pages)

Return.

1. Strong invariants and variants expressible in d$\mathcal{L}$
2. d$\mathcal{L}$ expressible in FOD
3. valid d$\mathcal{L}$ formulas d$\mathcal{L}$-derivable from corresponding FOD axioms
4. finite FOD formula characterising unbounded hybrid repetition
5. FOD characterises $\mathbb{R}$-Gödel encoding
6. First-order expressible & program rendition: $\forall\phi\ \exists F \in \text{FOD} \quad \vDash \phi \leftrightarrow F$
7. Propositionally & first-order complete
8. Relative complete for first-order safety $F \rightarrow [\alpha]G$
9. Relative complete for first-order liveness $F \rightarrow \langle\alpha\rangle G$

$\square$

Relative Completeness Proof

$$\vDash \phi \quad \text{iff} \quad Taut_{\mathsf{FOD}} \vdash \phi$$

$$\text{where} \quad \mathsf{FOD} = \mathsf{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$$

Proof (Relative Completeness, 10 pages)

Return .

1. Strong invariants and variants expressible in d$\mathcal{L}$
2. d$\mathcal{L}$ expressible in FOD
3. valid d$\mathcal{L}$ formulas d$\mathcal{L}$-derivable from corresponding FOD axioms
4. finite FOD formula characterising unbounded hybrid repetition
5. FOD characterises $\mathbb{R}$-Gödel encoding
6. First-order expressible & program rendition: $\forall \phi \ \exists F \in \mathsf{FOD} \quad \vDash \phi \leftrightarrow F$
7. Propositionally & first-order complete
8. Relative complete for first-order safety $F \rightarrow [\alpha]G$
9. Relative complete for first-order liveness $F \rightarrow \langle \alpha \rangle G$

□

Relative Completeness Proof

$$\vDash \phi \quad \text{iff} \quad Taut_{\mathrm{FOD}} \vdash \phi$$

$$\text{where} \quad \mathrm{FOD} = \mathrm{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \dots, x_n' = \theta_n]F$$

Proof (Relative Completeness, 10 pages) Return

1. Strong invariants and variants expressible in $d\mathcal{L}$
2. $d\mathcal{L}$ expressible in FOD
3. valid $d\mathcal{L}$ formulas $d\mathcal{L}$-derivable from corresponding FOD axioms
4. finite FOD formula characterising unbounded hybrid repetition
5. FOD characterises $\mathbb{R}$-Gödel encoding
6. First-order expressible & program rendition: $\forall \phi\ \exists F \in \mathrm{FOD} \quad \vDash \phi \leftrightarrow F$
7. Propositionally & first-order complete
8. Relative complete for first-order safety $F \rightarrow [\alpha]G$
9. Relative complete for first-order liveness $F \rightarrow \langle\alpha\rangle G$

□

Relative Completeness Proof

$$\vDash \phi \quad \text{iff} \quad Taut_{\text{FOD}} \vdash \phi$$

$$\text{where} \quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$$

Proof (Relative Completeness, 10 pages)

Return .

1. Strong invariants and variants expressible in d$\mathcal{L}$
2. d$\mathcal{L}$ expressible in FOD
3. valid d$\mathcal{L}$ formulas d$\mathcal{L}$-derivable from corresponding FOD axioms
4. finite FOD formula characterising unbounded hybrid repetition
5. FOD characterises $\mathbb{R}$-Gödel encoding
6. First-order expressible & program rendition: $\forall \phi \; \exists F \in \text{FOD} \quad \vDash \phi \leftrightarrow F$
7. Propositionally & first-order complete
8. Relative complete for first-order safety $F \rightarrow [\alpha]G$
9. Relative complete for first-order liveness $F \rightarrow \langle \alpha \rangle G$

□

Relative Completeness Proof

$$\vDash \phi \quad \text{iff} \quad Taut_{\mathsf{FOD}} \vdash \phi$$

$$\text{where} \quad \mathsf{FOD} = \mathsf{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$$

Proof (Relative Completeness, 10 pages)

Return.

1. Strong invariants and variants expressible in d$\mathcal{L}$
2. d$\mathcal{L}$ expressible in FOD
3. valid d$\mathcal{L}$ formulas d$\mathcal{L}$-derivable from corresponding FOD axioms
4. finite FOD formula characterising unbounded hybrid repetition
5. FOD characterises $\mathbb{R}$-Gödel encoding
6. First-order expressible & program rendition: $\forall \phi\ \exists F \in \mathsf{FOD} \quad \vDash \phi \leftrightarrow F$
7. Propositionally & first-order complete
8. Relative complete for first-order safety $F \to [\alpha]G$
9. Relative complete for first-order liveness $F \to \langle\alpha\rangle G$

□

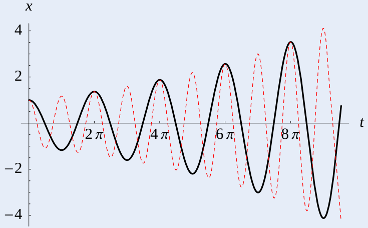
Relative Completeness Proof

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

Return

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$



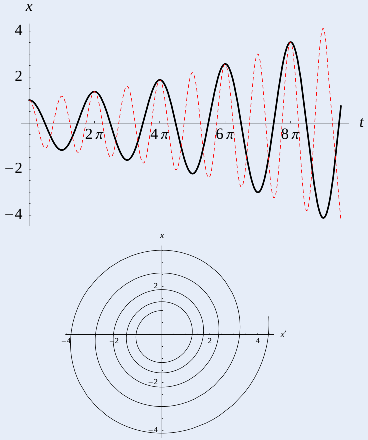
Relative Completeness Proof

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

◂ Return

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$



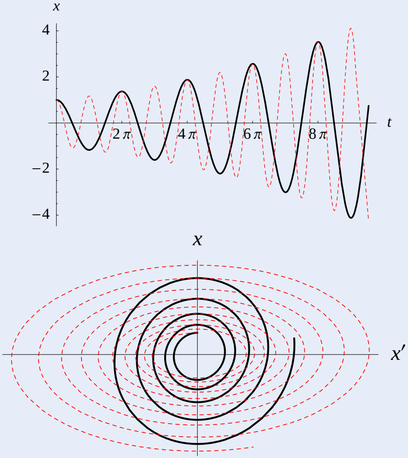
Relative Completeness Proof

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

◂ Return

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$

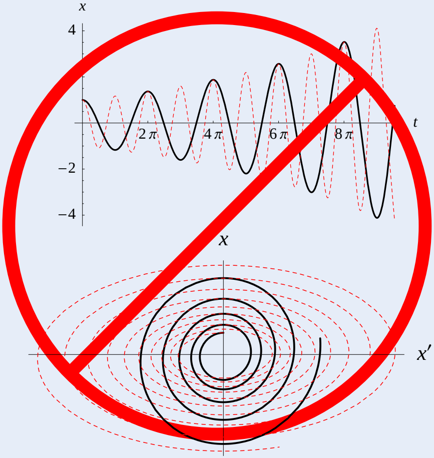


where $\text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \dots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

Return.

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$ not differentiable!



Relative Completeness Proof

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \ldots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

Return

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$

$$\sum_{i=1}^{\infty} \frac{a_i}{2^i} = 0.a_1 a_2 \ldots$$

$$\sum_{i=1}^{\infty} \frac{b_i}{2^i} = 0.b_1 b_2 \ldots$$

$$\longleftrightarrow \quad \sum_{i=0}^{\infty} \left(\frac{a_i}{2^{2i+1}} + \frac{b_i}{2^{2i+2}} \right) = 0.a_1 b_1 a_2 b_2 \ldots$$

□

Relative Completeness Proof

where $\quad \text{FOD} = \text{FOL}_{\mathbb{R}} + [x_1' = \theta_1, \dots, x_n' = \theta_n]F$

Proof ($\mathbb{R}$-Gödel encoding)

◂ Return

FOD characterises constructive bijection $\mathbb{R} \to \mathbb{R}^2$

$$\sum_{i=1}^{\infty} \frac{a_i}{2^i} = 0.a_1 a_2 \dots$$

$$\sum_{i=1}^{\infty} \frac{b_i}{2^i} = 0.b_1 b_2 \dots$$

$$\sum_{i=0}^{\infty} \left(\frac{a_i}{2^{2i+1}} + \frac{b_i}{2^{2i+2}} \right) = 0.a_1 b_1 a_2 b_2 \dots$$

$$2^n = z \quad \leftrightarrow \quad \langle x := 1; \tau := 0; x' = x \ln 2 \land \tau' = 1 \rangle (\tau = n \land x = z)$$

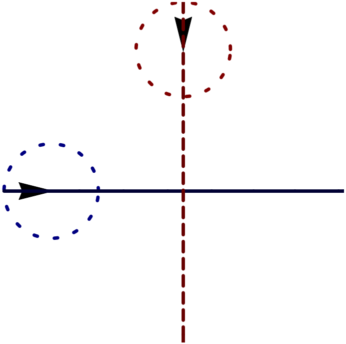
$$\ln 2 = z \quad \leftrightarrow \quad \langle x := 1; \tau := 0; x' = x \land \tau' = 1 \rangle (x = 2 \land \tau = z)$$

□

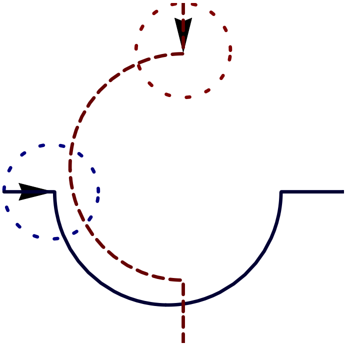
Outline Background

Outline Background

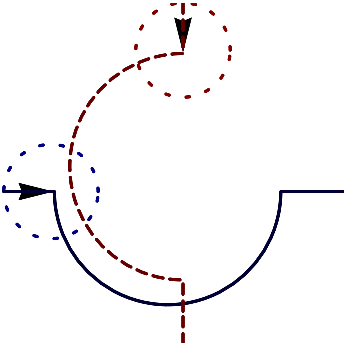
Air Traffic Control



Air Traffic Control



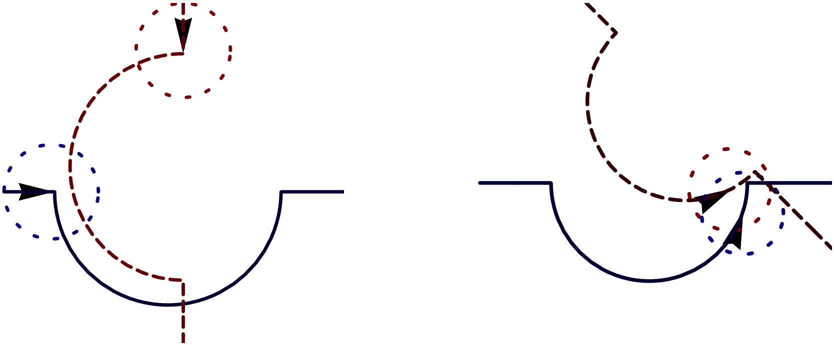
Air Traffic Control



Verification?

looks correct

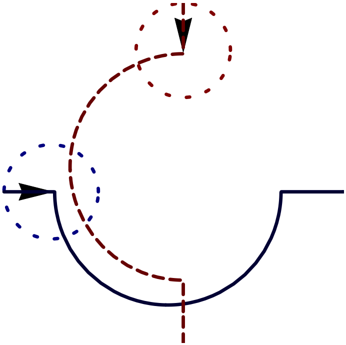
Air Traffic Control

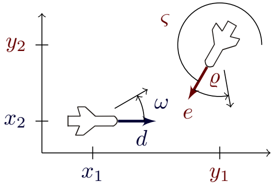


Verification?

looks correct NO!

Air Traffic Control



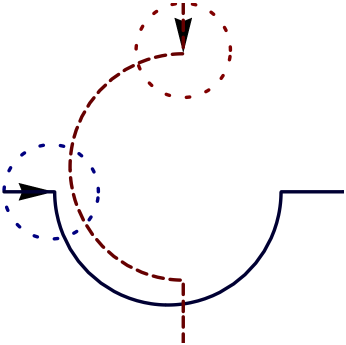


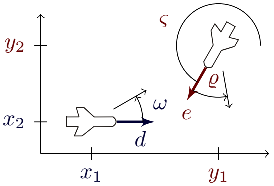
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

Verification?

looks correct NO!

Air Traffic Control





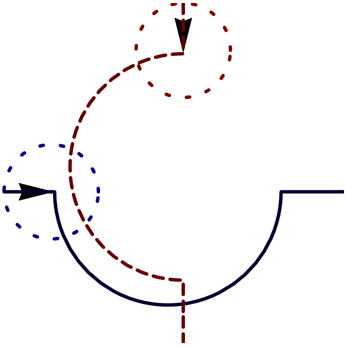
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

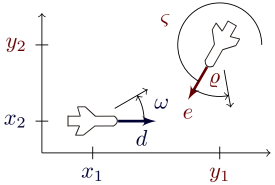
Example ("Solving" differential equations)

$$x_1(t) = \frac{1}{\omega\varpi}\big(x_1\omega\varpi\cos t\omega - v_2\omega\cos t\omega\sin\vartheta + v_2\omega\cos t\omega\cos t\varpi\sin\vartheta - v_1\varpi\sin t\omega$$

$$+ x_2\omega\varpi\sin t\omega - v_2\omega\cos\vartheta\cos t\varpi\sin t\omega - v_2\omega\sqrt{1-\sin\vartheta^2}\sin t\omega$$

$$+ v_2\omega\cos\vartheta\cos t\omega\sin t\varpi + v_2\omega\sin\vartheta\sin t\omega\sin t\varpi\big)\ldots$$





$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

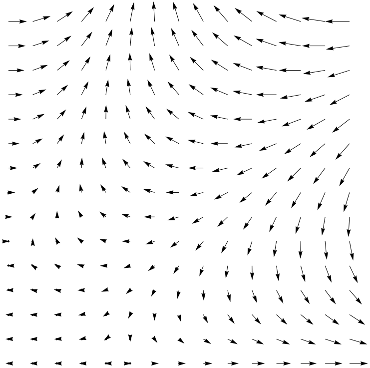
Example ("Solving" differential equations)

$$\forall t \geq 0 \quad \frac{1}{\omega\varpi}\big(x_1\omega\varpi \cos t\omega - v_2\omega \cos t\omega \sin\vartheta + v_2\omega \cos t\omega \cos t\varpi \sin\vartheta - v_1\varpi \sin t\omega$$
$$+ x_2\omega\varpi \sin t\omega - v_2\omega \cos\vartheta \cos t\varpi \sin t\omega - v_2\omega\sqrt{1 - \sin\vartheta^2} \sin t\omega$$
$$+ v_2\omega \cos\vartheta \cos t\omega \sin t\varpi + v_2\omega \sin\vartheta \sin t\omega \sin t\varpi\big) \ldots$$

"Definition" (Differential Invariant)

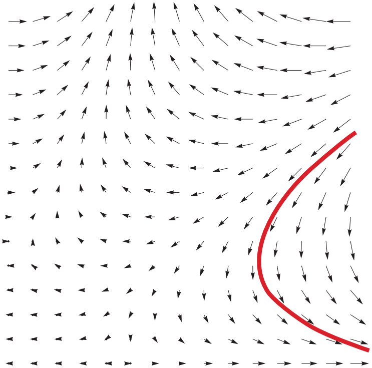
Details

"Formula that remains true in the direction of the dynamics"



"Definition" (Differential Invariant) ▸ Details

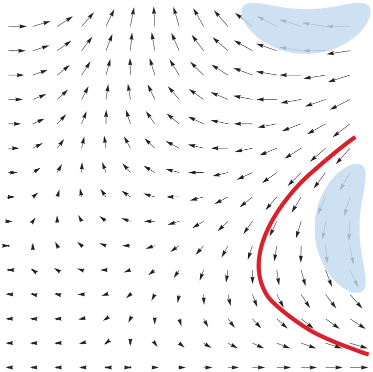
"Formula that remains true in the direction of the dynamics"



"Definition" (Differential Invariant)

Details

"Formula that remains true in the direction of the dynamics"



Definition (Differential Invariant) ▸ Details

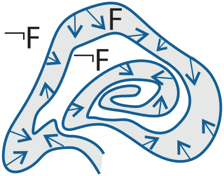
F closed under total differentiation with respect to differential constraints

André Platzer.
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 35(1): 309–352, 2010.

Definition (Differential Invariant)

Details

F closed under total differentiation with respect to differential constraints



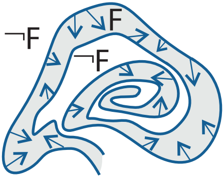
$$\frac{(\chi \to F')}{\chi \to F \to [x' = \theta \,\&\, \chi]F}$$

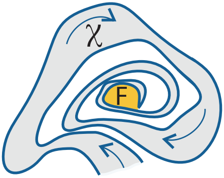
$$\frac{F \to [\alpha]F}{F \to [\alpha^*]F}$$

Definition (Differential Invariant)

Details

F closed under total differentiation with respect to differential constraints





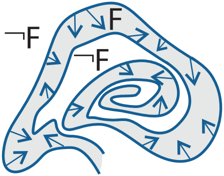
$$\frac{(\chi \to F')}{\chi \to F \to [x' = \theta \,\&\, \chi]F}$$

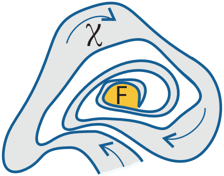
Differential Induction: Local Dynamics w/o Solutions

Definition (Differential Invariant)

Details

F closed under total differentiation with respect to differential constraints





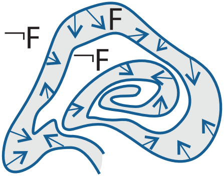
$$\frac{(\chi \to F')}{\chi \to F \to [x' = \theta \,\&\, \chi] F}$$

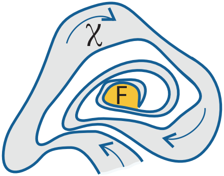
$$\frac{(\neg F \land \chi \to F'_{\gg})}{[x' = \theta \,\&\, \neg F]\chi \to \langle x' = \theta \,\&\, \chi\rangle F}$$

Differential Induction: Local Dynamics w/o Solutions

Definition (Differential Invariant) ▸ Details

F closed under total differentiation with respect to differential constraints



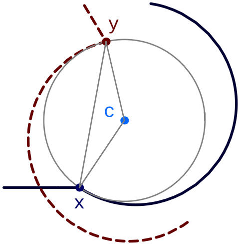


$$\frac{(\chi \rightarrow F')}{\chi \rightarrow F \rightarrow [x' = \theta \,\&\, \chi]F}$$

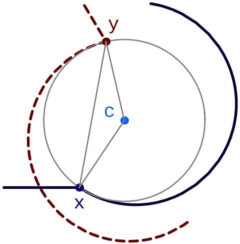
$$\frac{(\neg F \land \chi \rightarrow F'_{\gg})}{[x' = \theta \,\&\, \neg F]\chi \rightarrow \langle x' = \theta \,\&\, \chi\rangle F}$$

Total differential F' of *formulas*?

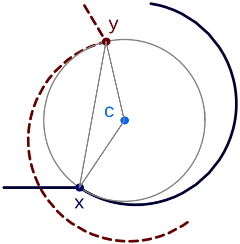
$$\overline{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



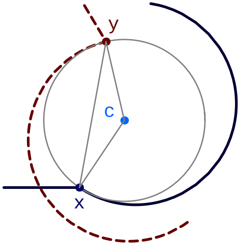
$$\dfrac{\dfrac{\partial\|x-y\|^2}{\partial x_1}x_1' + \dfrac{\partial\|x-y\|^2}{\partial y_1}y_1' + \dfrac{\partial\|x-y\|^2}{\partial x_2}x_2' + \dfrac{\partial\|x-y\|^2}{\partial y_2}y_2' \geq \dfrac{\partial p^2}{\partial x_1}x_1' \ldots}{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



$$\frac{\frac{\partial\|x-y\|^2}{\partial x_1}x_1' + \frac{\partial\|x-y\|^2}{\partial y_1}y_1' + \frac{\partial\|x-y\|^2}{\partial x_2}x_2' + \frac{\partial\|x-y\|^2}{\partial y_2}y_2' \geq \frac{\partial p^2}{\partial x_1}x_1' \ldots}{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



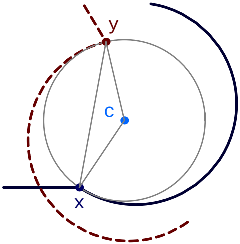
$$\frac{\frac{\partial\|x-y\|^2}{\partial x_1} d_1 + \frac{\partial\|x-y\|^2}{\partial y_1} e_1 + \frac{\partial\|x-y\|^2}{\partial x_2} d_2 + \frac{\partial\|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \dots}{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$

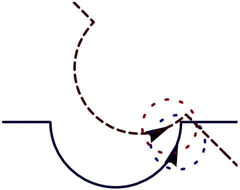


$$2(x_1 - y_1)(d_1 - e_1) + 2(x_2 - y_2)(d_2 - e_2) \geq 0$$

$$\frac{\partial \|x-y\|^2}{\partial x_1} d_1 + \frac{\partial \|x-y\|^2}{\partial y_1} e_1 + \frac{\partial \|x-y\|^2}{\partial x_2} d_2 + \frac{\partial \|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \ldots$$

$$[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2$$

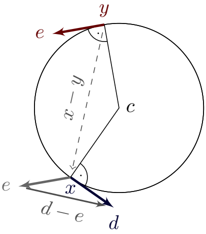


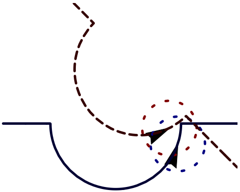


$$2(x_1 - y_1)(d_1 - e_1) + 2(x_2 - y_2)(d_2 - e_2) \geq 0$$

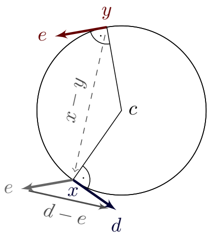
$$\frac{\partial \|x-y\|^2}{\partial x_1} d_1 + \frac{\partial \|x-y\|^2}{\partial y_1} e_1 + \frac{\partial \|x-y\|^2}{\partial x_2} d_2 + \frac{\partial \|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \ldots$$

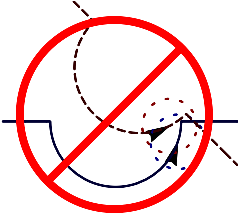
$$[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2$$





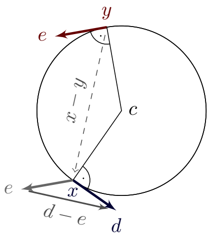
$$\frac{2(x_1 - y_1)(d_1 - e_1) + 2(x_2 - y_2)(d_2 - e_2) \geq 0}{\frac{\frac{\partial\|x-y\|^2}{\partial x_1} d_1 + \frac{\partial\|x-y\|^2}{\partial y_1} e_1 + \frac{\partial\|x-y\|^2}{\partial x_2} d_2 + \frac{\partial\|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \dots}{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}}$$





$$\overline{\quad .. \rightarrow [d_1' = -\omega d_2, e_1' = -\omega e_2, x_2' = d_2, d_2' = \omega d_1, ..] d_1 - e_1 = -\omega(x_2 - y_2)}$$

$$\frac{\frac{\frac{2(x_1-y_1)(-\omega(x_2-y_2))+2(x_2-y_2)\omega(x_1-y_1)\geq 0}{2(x_1-y_1)(d_1-e_1)+2(x_2-y_2)(d_2-e_2)\geq 0}}{\frac{\partial\|x-y\|^2}{\partial x_1}d_1+\frac{\partial\|x-y\|^2}{\partial y_1}e_1+\frac{\partial\|x-y\|^2}{\partial x_2}d_2+\frac{\partial\|x-y\|^2}{\partial y_2}e_2\geq\frac{\partial p^2}{\partial x_1}d_1\dots}}{[x_1'=d_1,d_1'=-\omega d_2,x_2'=d_2,d_2'=\omega d_1,..](x_1-y_1)^2+(x_2-y_2)^2\geq p^2}$$



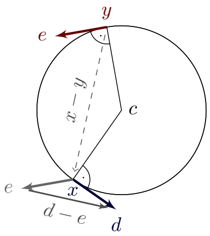
$$\overline{..\to[d_1'=-\omega d_2,e_1'=-\omega e_2,x_2'=d_2,d_2'=\omega d_1,..]d_1-e_1=-\omega(x_2-y_2)}$$

$$2(x_1 - y_1)(-\omega(x_2 - y_2)) + 2(x_2 - y_2)\omega(x_1 - y_1) \geq 0$$

$$2(x_1 - y_1)(d_1 - e_1) + 2(x_2 - y_2)(d_2 - e_2) \geq 0$$

$$\frac{\partial\|x-y\|^2}{\partial x_1} d_1 + \frac{\partial\|x-y\|^2}{\partial y_1} e_1 + \frac{\partial\|x-y\|^2}{\partial x_2} d_2 + \frac{\partial\|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \ldots$$

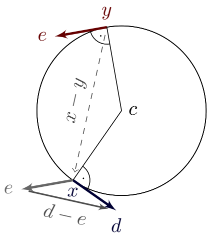
$$[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2$$



$$\frac{\partial(d_1 - e_1)}{\partial d_1} d_1' + \frac{\partial(d_1 - e_1)}{\partial e_1} e_1' = -\frac{\partial\omega(x_2 - y_2)}{\partial x_2} x_2' - \frac{\partial\omega(x_2 - y_2)}{\partial y_2} y_2'$$

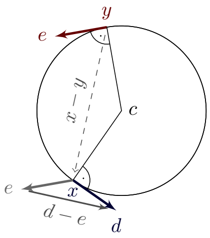
$$.. \to [d_1' = -\omega d_2, e_1' = -\omega e_2, x_2' = d_2, d_2' = \omega d_1, ..]\, d_1 - e_1 = -\omega(x_2 - y_2)$$

$$\frac{\dfrac{\dfrac{2(x_1-y_1)(-\omega(x_2-y_2))+2(x_2-y_2)\omega(x_1-y_1)\geq 0}{2(x_1-y_1)(d_1-e_1)+2(x_2-y_2)(d_2-e_2)\geq 0}}{\frac{\partial\|x-y\|^2}{\partial x_1}d_1+\frac{\partial\|x-y\|^2}{\partial y_1}e_1+\frac{\partial\|x-y\|^2}{\partial x_2}d_2+\frac{\partial\|x-y\|^2}{\partial y_2}e_2\geq\frac{\partial p^2}{\partial x_1}d_1\ldots}}{[x_1'=d_1,d_1'=-\omega d_2,x_2'=d_2,d_2'=\omega d_1,..](x_1-y_1)^2+(x_2-y_2)^2\geq p^2}$$



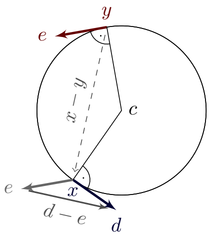
$$\frac{\frac{\partial(d_1-e_1)}{\partial d_1}d_1'+\frac{\partial(d_1-e_1)}{\partial e_1}e_1'=-\frac{\partial\omega(x_2-y_2)}{\partial x_2}x_2'-\frac{\partial\omega(x_2-y_2)}{\partial y_2}y_2'}{..\to[d_1'=-\omega d_2,e_1'=-\omega e_2,x_2'=d_2,d_2'=\omega d_1,..]d_1-e_1=-\omega(x_2-y_2)}$$

$$\frac{2(x_1 - y_1)(-\omega(x_2 - y_2)) + 2(x_2 - y_2)\omega(x_1 - y_1) \geq 0}{\dfrac{2(x_1 - y_1)(d_1 - e_1) + 2(x_2 - y_2)(d_2 - e_2) \geq 0}{\dfrac{\frac{\partial\|x-y\|^2}{\partial x_1} d_1 + \frac{\partial\|x-y\|^2}{\partial y_1} e_1 + \frac{\partial\|x-y\|^2}{\partial x_2} d_2 + \frac{\partial\|x-y\|^2}{\partial y_2} e_2 \geq \frac{\partial p^2}{\partial x_1} d_1 \ldots}{[x_1' = d_1, d_1' = -\omega d_2, x_2' = d_2, d_2' = \omega d_1, ..](x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}}}$$



$$\frac{\dfrac{\partial(d_1 - e_1)}{\partial d_1}(-\omega d_2) + \dfrac{\partial(d_1 - e_1)}{\partial e_1}(-\omega e_2) = -\dfrac{\partial\omega(x_2 - y_2)}{\partial x_2} d_2 - \dfrac{\partial\omega(x_2 - y_2)}{\partial y_2} e_2}{.. \to [d_1' = -\omega d_2, e_1' = -\omega e_2, x_2' = d_2, d_2' = \omega d_1, ..] d_1 - e_1 = -\omega(x_2 - y_2)}$$

$$\frac{\dfrac{\dfrac{2(x_1-y_1)(-\omega(x_2-y_2))+2(x_2-y_2)\omega(x_1-y_1)\geq 0}{2(x_1-y_1)(d_1-e_1)+2(x_2-y_2)(d_2-e_2)\geq 0}}{\frac{\partial\|x-y\|^2}{\partial x_1}d_1+\frac{\partial\|x-y\|^2}{\partial y_1}e_1+\frac{\partial\|x-y\|^2}{\partial x_2}d_2+\frac{\partial\|x-y\|^2}{\partial y_2}e_2\geq\frac{\partial p^2}{\partial x_1}d_1\dots}}{[x_1'=d_1,d_1'=-\omega d_2,x_2'=d_2,d_2'=\omega d_1,..](x_1-y_1)^2+(x_2-y_2)^2\geq p^2}$$



$$\frac{\dfrac{-\omega d_2+\omega e_2=-\omega(d_2-e_2)}{\frac{\partial(d_1-e_1)}{\partial d_1}(-\omega d_2)+\frac{\partial(d_1-e_1)}{\partial e_1}(-\omega e_2)=-\frac{\partial\omega(x_2-y_2)}{\partial x_2}d_2-\frac{\partial\omega(x_2-y_2)}{\partial y_2}e_2}}{..\to[d_1'=-\omega d_2,e_1'=-\omega e_2,x_2'=d_2,d_2'=\omega d_1,..]d_1-e_1=-\omega(x_2-y_2)}$$

$$\dfrac{\dfrac{\dfrac{2(x_1-y_1)(-\omega(x_2-y_2))+2(x_2-y_2)\omega(x_1-y_1)\geq 0}{2(x_1-y_1)(d_1-e_1)+2(x_2-y_2)(d_2-e_2)\geq 0}}{\frac{\partial\|x-y\|^2}{\partial x_1}d_1+\frac{\partial\|x-y\|^2}{\partial y_1}e_1+\frac{\partial\|x-y\|^2}{\partial x_2}d_2+\frac{\partial\|x-y\|^2}{\partial y_2}e_2\geq\frac{\partial p^2}{\partial x_1}d_1\dots}}{[x_1'=d_1,d_1'=-\omega d_2,x_2'=d_2,d_2'=\omega d_1,..](x_1-y_1)^2+(x_2-y_2)^2\geq p^2}$$

Proposition (Differential cut saturation)

F differential invariant of $[x'=\theta\,\&\,H]\phi$, then

$$[x'=\theta\,\&\,H]\phi \quad\text{iff}\quad [x'=\theta\,\&\,H\wedge F]\phi$$

$$\dfrac{\dfrac{-\omega d_2+\omega e_2=-\omega(d_2-e_2)}{\frac{\partial(d_1-e_1)}{\partial d_1}(-\omega d_2)+\frac{\partial(d_1-e_1)}{\partial e_1}(-\omega e_2)=-\frac{\partial\omega(x_2-y_2)}{\partial x_2}d_2-\frac{\partial\omega(x_2-y_2)}{\partial y_2}e_2}}{..\to[d_1'=-\omega d_2,e_1'=-\omega e_2,x_2'=d_2,d_2'=\omega d_1,..]d_1-e_1=-\omega(x_2-y_2)}$$

$$\dfrac{\dfrac{\dfrac{2(x_1-y_1)(-\omega(x_2-y_2))+2(x_2-y_2)\omega(x_1-y_1)\geq 0}{2(x_1-y_1)(d_1-e_1)+2(x_2-y_2)(d_2-e_2)\geq 0}}{\frac{\partial\|x-y\|^2}{\partial x_1}d_1+\frac{\partial\|x-y\|^2}{\partial y_1}e_1+\frac{\partial\|x-y\|^2}{\partial x_2}d_2+\frac{\partial\|x-y\|^2}{\partial y_2}e_2\geq\frac{\partial p^2}{\partial x_1}d_1\dots}}{[x_1'=d_1,d_1'=-\omega d_2,x_2'=d_2,d_2'=\omega d_1,..](x_1-y_1)^2+(x_2-y_2)^2\geq p^2}$$

refine dynamics

by differential cut

$$\dfrac{\dfrac{-\omega d_2+\omega e_2=-\omega(d_2-e_2)}{\frac{\partial(d_1-e_1)}{\partial d_1}(-\omega d_2)+\frac{\partial(d_1-e_1)}{\partial e_1}(-\omega e_2)=-\frac{\partial\omega(x_2-y_2)}{\partial x_2}d_2-\frac{\partial\omega(x_2-y_2)}{\partial y_2}e_2}}{..\rightarrow[d_1'=-\omega d_2,e_1'=-\omega e_2,x_2'=d_2,d_2'=\omega d_1,..]d_1-e_1=-\omega(x_2-y_2)}$$

Counterexample

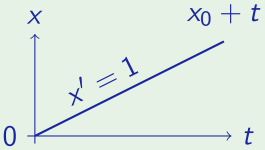
$$\frac{x^2 \leq 0 \rightarrow 2x \cdot 1 \leq 0}{x^2 \leq 0 \rightarrow [x' = 1]x^2 \leq 0}$$

$$\frac{x > 0 \rightarrow -x < 0}{\langle x' = -x \rangle x \leq 0}$$

$$\frac{x' \neq 0}{x \neq 5 \rightarrow [x' = 1]x \neq 5}$$

Counterexample

$$\frac{x^2 \leq 0 \rightarrow 2x \cdot 1 \leq 0}{x^2 \leq 0 \rightarrow [x' = 1]x^2 \leq 0}$$



$$\frac{x > 0 \rightarrow -x < 0}{\langle x' = -x \rangle x \leq 0}$$

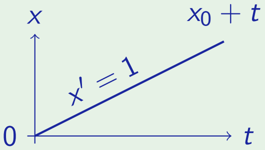
$$\frac{x' \neq 0}{x \neq 5 \rightarrow [x' = 1]x \neq 5}$$

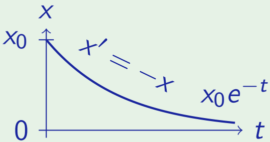
Counterexample

$$\frac{x^2 \leq 0 \rightarrow 2x \cdot 1 \leq 0}{x^2 \leq 0 \rightarrow [x' = 1]x^2 \leq 0}$$

$$\frac{x > 0 \rightarrow -x < 0}{\langle x' = -x \rangle x \leq 0}$$

$$\frac{x' \neq 0}{x \neq 5 \rightarrow [x' = 1]x \neq 5}$$

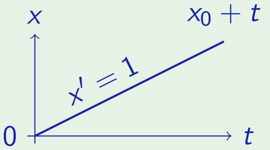




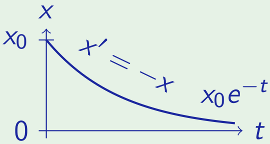
Differential Invariants and Variants

Counterexample

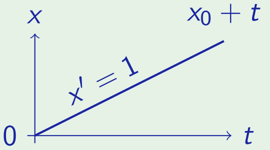
$$\frac{x^2 \leq 0 \rightarrow 2x \cdot 1 \leq 0}{x^2 \leq 0 \rightarrow [x' = 1]x^2 \leq 0}$$



$$\frac{x > 0 \rightarrow -x < 0}{\langle x' = -x \rangle x \leq 0}$$



$$\frac{x' \neq 0}{x \neq 5 \rightarrow [x' = 1]x \neq 5}$$



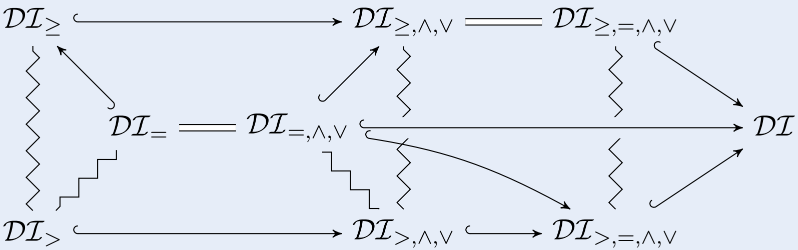
Outline Background

The Structure of Differential Invariants

Theorem (Closure properties of differential invariants)

Closed under conjunction, differentiation, and propositional equivalences.

Theorem (Differential Invariance Chart)



André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Differential Cuts

$$\frac{F \rightarrow [x' = \theta \,\&\, H]C \qquad F \rightarrow [x' = \theta \,\&\, (H \land C)]F}{F \rightarrow [x' = \theta \,\&\, H]F}$$

André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Differential Cuts

$$\frac{F \rightarrow [x' = \theta \,\&\, H]C \qquad F \rightarrow [x' = \theta \,\&\, (H \land C)]F}{F \rightarrow [x' = \theta \,\&\, H]F}$$

André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Differential Cuts

$$\frac{F \rightarrow [x' = \theta \,\&\, H]C \qquad F \rightarrow [x' = \theta \,\&\, (H \wedge C)]F}{F \rightarrow [x' = \theta \,\&\, H]F}$$

Theorem (Gentzen's Cut Elimination)

$$\frac{A \rightarrow B \vee C \qquad A \wedge C \rightarrow B}{A \rightarrow B} \qquad \textit{cut can be eliminated}$$

André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Differential Cuts

$$\frac{F \to [x' = \theta \,\&\, H]C \qquad F \to [x' = \theta \,\&\, (H \land C)]F}{F \to [x' = \theta \,\&\, H]F}$$

Theorem (Gentzen's Cut Elimination)

$$\frac{A \to B \lor C \quad A \land C \to B}{A \to B}$$ *cut can be eliminated*

Theorem (No Differential Cut Elimination)

Deductive power with differential cut exceeds deductive power without.

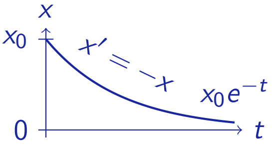
$$\mathcal{DCI} > \mathcal{DI}$$

André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Ex: Exponentials

Counterexample ()

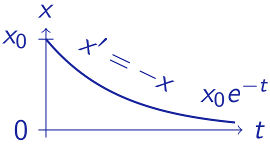
$$\overline{x > 0 \rightarrow [x' = -x]x > 0}$$



Ex: Exponentials

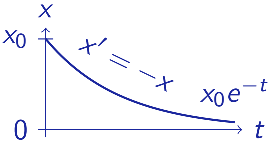
Counterexample ()

$$\dfrac{\overline{x' > 0}}{x > 0 \rightarrow [x' = -x]x > 0}$$



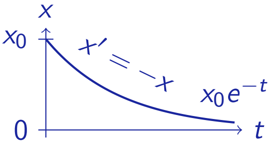
Counterexample ()

$$\cfrac{\cfrac{\cfrac{}{-x > 0}}{x' > 0}}{x > 0 \rightarrow [x' = -x]x > 0}$$



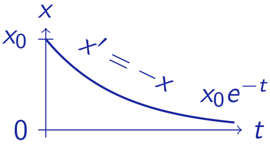
Counterexample (Cannot prove)

$$\cfrac{\cfrac{\cfrac{\text{not valid}}{-x > 0}}{x' > 0}}{x > 0 \rightarrow [x' = -x]x > 0}$$



Example (Successful proof)

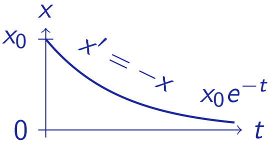
$$\frac{}{x > 0 \rightarrow [x' = -x]x > 0}$$



Differential Auxiliaries

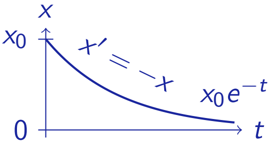
Example (Successful proof)

$$\dfrac{\overline{x > 0 \leftrightarrow \exists y\, xy^2 = 1} \qquad \overline{xy^2 = 1 \rightarrow [x' = -x, y' = \frac{y}{2}]xy^2 = 1}}{x > 0 \rightarrow [x' = -x]x > 0}$$



Example (Successful proof)

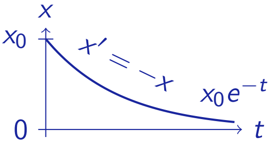
$$\dfrac{\dfrac{*}{x > 0 \leftrightarrow \exists y\, xy^2 = 1} \qquad \dfrac{}{xy^2 = 1 \rightarrow [x' = -x, y' = \frac{y}{2}]xy^2 = 1}}{x > 0 \rightarrow [x' = -x]x > 0}$$



Differential Auxiliaries

Example (Successful proof)

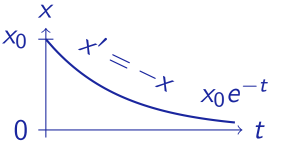
$$
\cfrac{\cfrac{*}{x > 0 \leftrightarrow \exists y\, xy^2 = 1} \qquad \cfrac{\cfrac{}{{\color{red}x'}y^2 + x2y{\color{red}y'} = 0}}{xy^2 = 1 \rightarrow [x' = -x, y' = \frac{y}{2}]xy^2 = 1}}{x > 0 \rightarrow [x' = -x]x > 0}
$$



Differential Auxiliaries

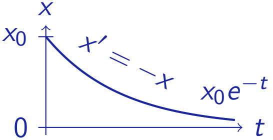
Example (Successful proof)

$$\dfrac{\dfrac{*}{x>0 \leftrightarrow \exists y\, xy^2=1} \qquad \dfrac{\dfrac{\dfrac{}{-xy^2+2xy\frac{y}{2}=0}}{x'y^2+x2yy'=0}}{xy^2=1 \to [x'=-x,\, y'=\frac{y}{2}]xy^2=1}}{x>0 \to [x'=-x]x>0}$$



Example (Successful proof)

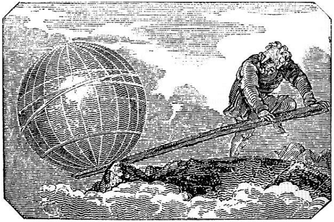
$$\dfrac{\dfrac{*}{x>0 \leftrightarrow \exists y\, xy^2=1} \qquad \dfrac{\dfrac{\dfrac{*}{-xy^2+2xy\frac{y}{2}=0}}{x'y^2+x2yy'=0}}{xy^2=1 \rightarrow [x'=-x,\, y'=\frac{y}{2}]xy^2=1}}{x>0 \rightarrow [x'=-x]x>0}$$

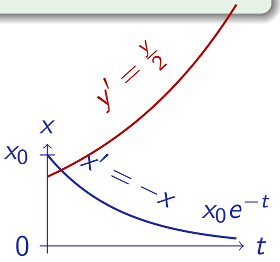


Differential Auxiliaries

Example (Successful proof)

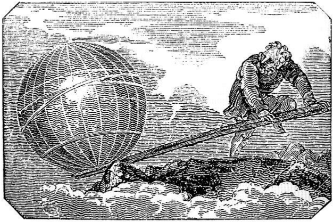
$$\dfrac{\dfrac{*}{x>0 \leftrightarrow \exists y\, xy^2=1} \qquad \dfrac{\dfrac{\dfrac{*}{-xy^2+2xy\frac{y}{2}=0}}{x'y^2+x2yy'=0}}{xy^2=1 \rightarrow [x'=-x,\, y'=\frac{y}{2}]xy^2=1}}{x>0 \rightarrow [x'=-x]x>0}$$

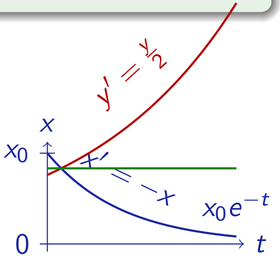




Example (Successful proof)

$$\dfrac{\dfrac{*}{x>0 \leftrightarrow \exists y\, xy^2=1} \qquad \dfrac{\dfrac{\dfrac{*}{-xy^2+2xy\frac{y}{2}=0}}{x'y^2+x2yy'=0}}{xy^2=1 \rightarrow [x'=-x, y'=\frac{y}{2}]xy^2=1}}{x>0 \rightarrow [x'=-x]x>0}$$





Differential Auxiliaries

$$\frac{\phi \leftrightarrow \exists y\, \psi \quad \psi \rightarrow [x' = \theta, y' = \vartheta \,\&\, H]\psi}{\phi \rightarrow [x' = \theta \,\&\, H]\phi}$$

if $y' = \vartheta$ has solution $y : [0, \infty) \to \mathbb{R}^n$

Theorem (Auxiliary Differential Variables)

Deductive power with differential auxiliaries exceeds deductive power without.

$$\mathcal{DCI} + DA > \mathcal{DCI}$$

André Platzer.
The structure of differential invariants and differential cut elimination.
Logical Methods in Computer Science, 2012.

Outline Background

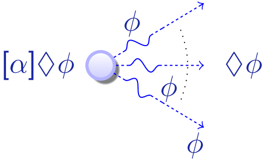
Temporal Modalities + Dynamic Modalities

problem	technique	Op	Par	T	closed
$train \models z < M$	TL-MC	✓	×	✓	×
$\models (\mathrm{Ax}(train) \rightarrow z < M)$	TL-calculus	×	...	✓	...
$\models [train]\, z < M$	DL-calculus	✓	✓	×	✓
$\models [train]\Box\, z < M$	dTL-calculus	✓	✓	✓	✓

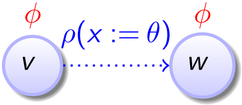
problem	technique	Op	Par	T	closed
$train \models z < M$	TL-MC	✓	×	✓	×
$\models (Ax(train) \to z < M)$	TL-calculus	×	...	✓	...
$\models [train]\, z < M$	DL-calculus	✓	✓	×	✓
$\models [train]\Box\, z < M$	dTL-calculus	✓	✓	✓	✓

differential temporal dynamic logic

$$dTL = TL + DL + HP$$

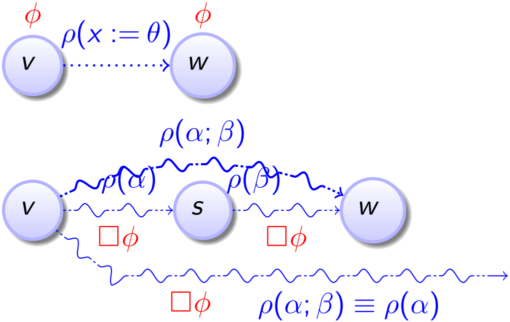


$$\frac{\phi \land [x := \theta]\phi}{[x := \theta]\Box\phi}$$



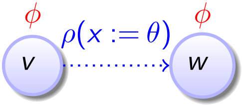
$$\frac{\phi \land [x := \theta]\phi}{[x := \theta]\Box\phi}$$

$$\frac{[\alpha]\Box\phi \land [\alpha][\beta]\Box\phi}{[\alpha;\beta]\Box\phi}$$

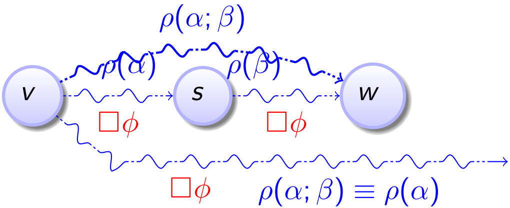


Proof Calculus for Temporal dTL

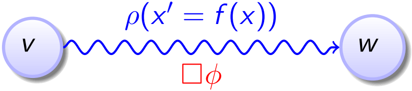
$$\frac{\phi \land [x := \theta]\phi}{[x := \theta]\Box\phi}$$



$$\frac{[\alpha]\Box\phi \land [\alpha][\beta]\Box\phi}{[\alpha;\beta]\Box\phi}$$

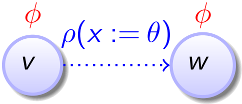


$$\frac{[x' = \theta]\phi}{[x' = \theta]\Box\phi}$$

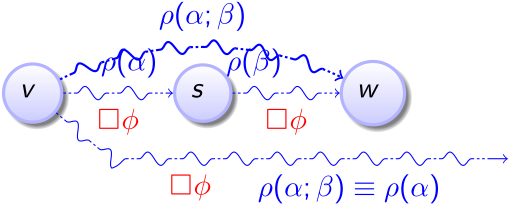


Proof Calculus for Temporal dTL

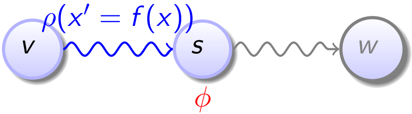
$$\frac{\phi \wedge [x := \theta]\phi}{[x := \theta]\Box\phi}$$



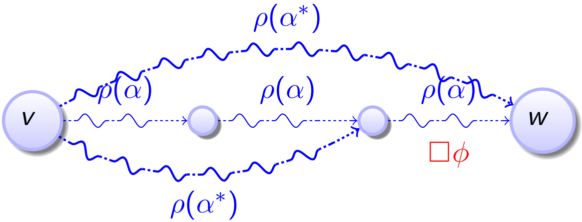
$$\frac{[\alpha]\Box\phi \wedge [\alpha][\beta]\Box\phi}{[\alpha;\beta]\Box\phi}$$



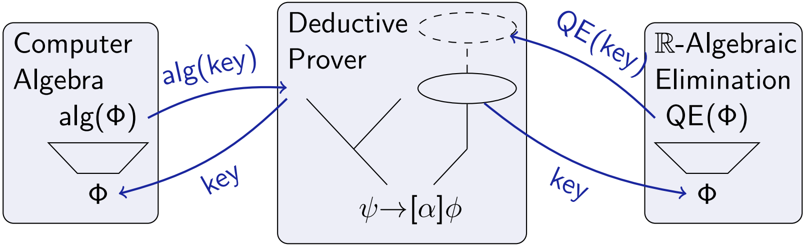
$$\frac{[x' = \theta]\phi}{[x' = \theta]\Box\phi}$$



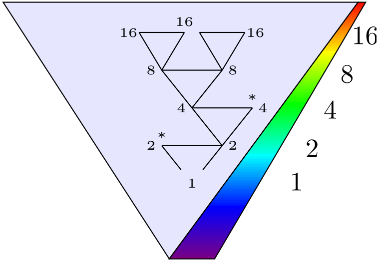
$$\frac{[\alpha^*][\alpha]\Box\phi}{[\alpha^*]\Box\phi}$$



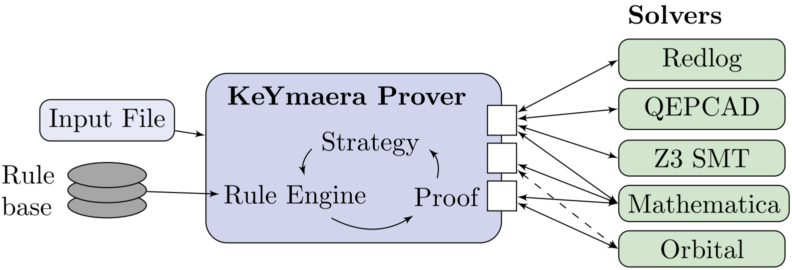
KeYmaera Verification Architecture



56 interactions?

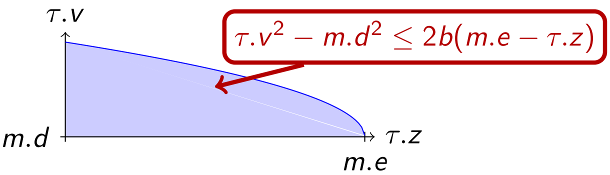


0–1 interactions!


Solvers
Input File
KeYmaera Prover
Strategy
Rule base
Rule Engine
Proof
Redlog
QEPCAD
Z3 SMT
Mathematica
Orbital

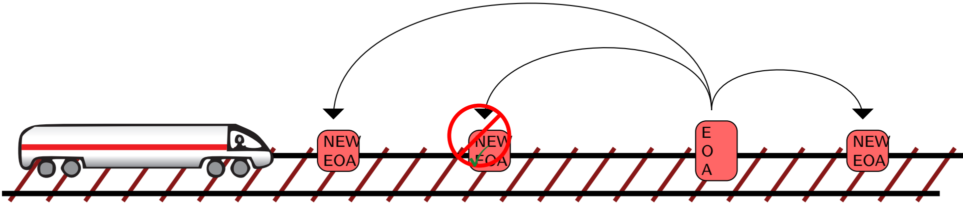
Outline Background

ETCS Controllability



Proposition (Controllability)

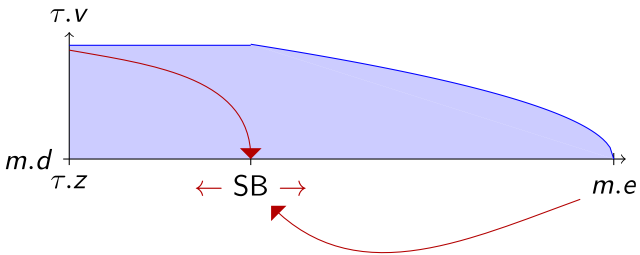
$$[\tau.z' = \tau.v, \tau.v' = -b \,\&\, \tau.v \geq 0](\tau.z \geq m.e \rightarrow \tau.v \leq m.d)$$
$$\equiv \tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z)$$



Proposition (RBC Controllability)

$$m.d \geq 0 \land b > 0 \rightarrow [m_0 := m;\ RBC] \Big($$
$$m_0.d^2 - m.d^2 \leq 2b(m.e - m_0.e) \land m_0.d \geq 0 \land m.d \geq 0 \leftrightarrow \forall \tau$$
$$((\langle m := m_0 \rangle \tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z)) \rightarrow \tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z)\Big)$$

ETCS Reactivity



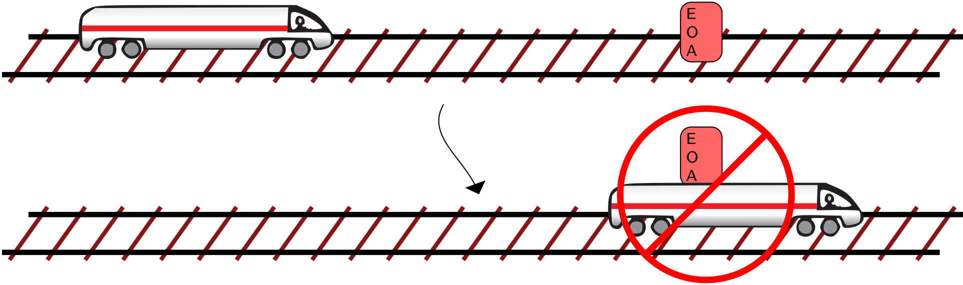
Proposition (Reactivity)

$$\Big(\forall m.e \, \forall \tau.z \, \big(m.e - \tau.z \geq SB \land \tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z) \rightarrow$$

$$[\tau.a := A;\ drive]\, \tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z)\big)\Big)$$

$$\equiv SB \geq \frac{\tau.v^2 - m.d^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon\, \tau.v\right)$$

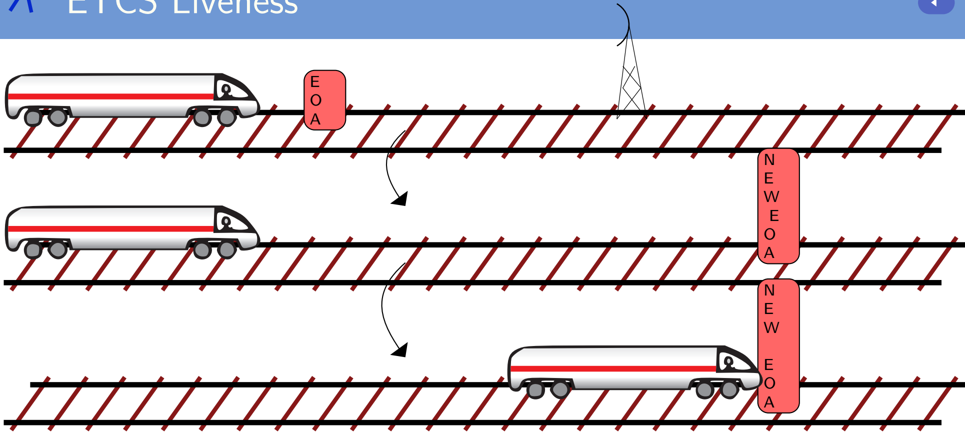
ETCS Safety



Proposition (Safety)

$$\tau.v^2 - m.d^2 \leq 2b(m.e - \tau.z) \rightarrow$$
$$[ETCS](\tau.z \geq m.e \rightarrow \tau.v \leq m.d)$$

ETCS Liveness



Proposition (Liveness)

$$\tau.v > 0 \land \varepsilon > 0 \rightarrow \forall P \, \langle ETCS \rangle \, \tau.z \geq P$$

Safety Despite Disturbances

So far: no wind, friction, etc.

Direct control of the acceleration

So far: no wind, friction, etc.

Direct control of the acceleration

Issue

This is unrealistic!

Safety Despite Disturbances

So far: no wind, friction, etc.

Direct control of the acceleration

Issue

This is unrealistic!

Solution Take disturbances into account.

Theorem

ETCS is controllable, reactive, and safe in the presence of disturbances.

So far: no wind, friction, etc.

Direct control of the acceleration

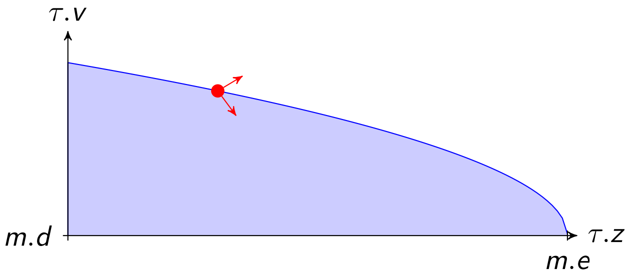
Issue

This is unrealistic!

Solution Take disturbances into account.

Theorem

ETCS is controllable, reactive, and safe in the presence of disturbances.



So far: no wind, friction, etc.

Direct control of the acceleration

Issue

This is unrealistic!

Solution Take disturbances into account.

Theorem

ETCS is controllable, reactive, and safe in the presence of disturbances.

Proof sketch

The system now contains $\tau.a - l \leq \tau.v' \leq \tau.a + u$ instead of $\tau.v' = \tau.a$.

$\leadsto$ We cannot solve the differential equations anymore.

$\leadsto$ Use differential invariants for approximation. For details see paper.

Platzer, A.:
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 35(1): 309–352, 2010.

Smooth Speed Control

So far

Almost completely non-deterministic control.

Smooth Speed Control

So far

Almost completely non-deterministic control.

Issue

This is unrealistic!

So far

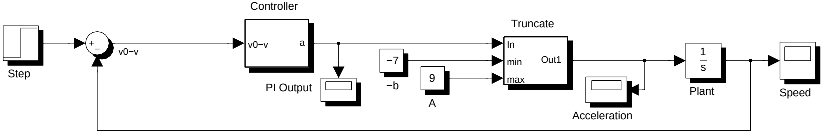
Almost completely non-deterministic control.

Issue

This is unrealistic!

Solution

Verify proportional-integral (PI) controllers used in trains.

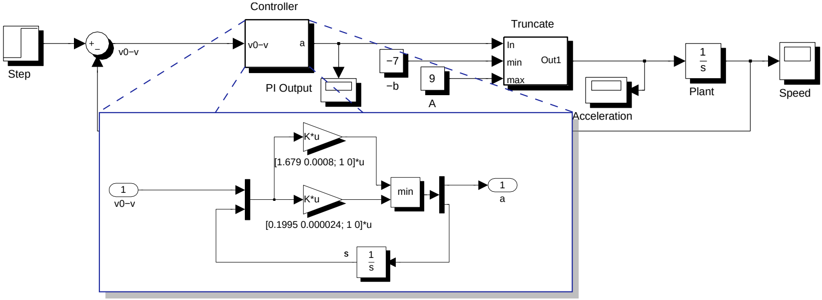


Smooth Speed Control

So far
Almost completely non-deterministic control.

Issue
This is unrealistic!

Solution Verify proportional-integral (PI) controllers used in trains.



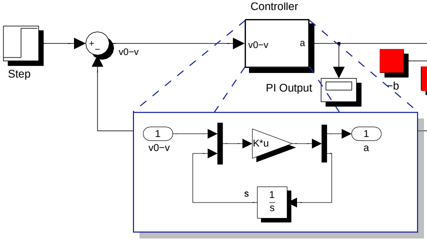
So far

Almost completely non-deterministic control.

Issue

This is unrealistic!

Solution Verify proportional-integral (PI) controllers used in trains.



Differential equation system

$$\tau.v' = \min\Big(A, \max\big(-b,\ \ell(\tau.v - m.r) - i\,s - c\,m.r\big)\Big) \land s' = \tau.v - m.r$$

Smooth Speed Control

So far

Almost completely non-deterministic control.

Issue

This is unrealistic!

Solution Verify proportional-integral (PI) controllers used in trains.

Theorem

The ETCS system remains safe when speed is controlled by a PI controller.

Proof sketch

Cannot solve differential equations really. Use differential invariants! For details see paper.

Platzer, A.:
Differential-algebraic dynamic logic for differential-algebraic programs.
J. Log. Comput., 35(1): 309–352, 2010.

Experimental Results (ETCS)

Case Study		Int	Time(s)	Mem(Mb)	Steps	Dim
controllability	train	0	0.6	6.9	14	5
controllability	RBC	0	0.5	6.4	42	12
controllability	RBC	0	0.9	6.5	82	12
reactivity		13	279.1	98.3	265	14
reactivity		0	103.9	61.7	47	14
safety		0	2052.4	204.3	153	14
liveness	essentials	4	35.2	92.2	62	10
liveness	simplified	6	9.6	23.5	134	13
controllability	disturbance	0	2.8	8.3	26	7
reactivity	disturbance	1	23.7	47.6	76	15
safety	disturbance	1	5805.2	34	218	16

provable automatically!

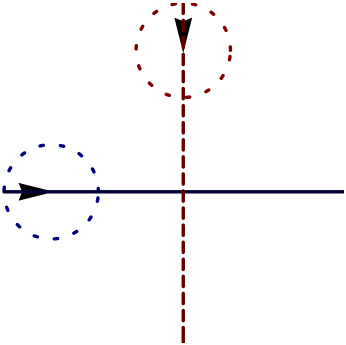
$$
\begin{aligned}
\text{spec} &: \tau.v^2 - \mathbf{m}.d^2 \le 2b(\mathbf{m}.e - \tau.p) \land \tau.v \ge 0 \land \mathbf{m}.d \ge 0 \land b > 0\\
&\quad \to \ [\text{ETCS}](\tau.p \ge \mathbf{m}.e \to \tau.v \le \mathbf{m}.d)\\
\text{ETCS} &: (\text{train} \cup \text{rbc})^*\\
\text{train} &: \text{spd};\ \text{atp};\ \text{move}\\
\text{spd} &: \ (?\tau.v \le \mathbf{m}.r;\ \tau.a := *;\ ?-b \le \tau.a \le A)\\
&\quad \cup(?\tau.v \ge \mathbf{m}.r;\ \tau.a := *;\ ?0 > \tau.a \ge -b)\\
\text{atp} &: SB := \frac{\tau.v^2 - \mathbf{m}.d^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon\, \tau.v\right);\\
&\quad (?(\mathbf{m}.e - \tau.p \le SB \lor rbc.message = emergency);\ \tau.a := -b)\\
&\quad \cup(?\mathbf{m}.e - \tau.p \ge SB \land rbc.message \ne emergency)\\
\text{move} &: t := 0;\ (\tau.p' = \tau.v, \tau.v' = \tau.a, t' = 1\,\&\,\tau.v \ge 0 \land t \le \varepsilon)\\
\text{rbc} &: \ (rbc.message := emergency)\\
&\quad \cup \big(\mathbf{m}_0 := \mathbf{m}; \mathbf{m} := *;\\
&\qquad ?\mathbf{m}.r \ge 0 \land \mathbf{m}.d \ge 0 \land \mathbf{m}_0.d^2 - \mathbf{m}.d^2 \le 2b(\mathbf{m}.e - \mathbf{m}_0.e)\big)
\end{aligned}
$$

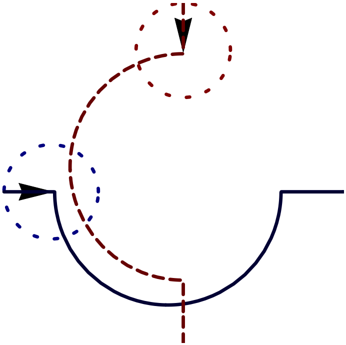
1 Branch of ETCS Proof

```
 state = 0,
 2 * b * (m - z) >= v ^ 2 - d ^ 2,
 v >= 0, d >= 0, v >= 0, ep >  0, b >  0, amax >  0, d >= 0
==>
    v <= vdes
 -> \forall R a_3;
      (   a_3 >= 0 & a_3 <= amax
       ->   (      m - z
               <= (amax / b + 1) * ep * v
                + (v ^ 2 - d ^ 2) / (2 * b)
                + (amax / b + 1) * amax * ep ^ 2 / 2
            -> \forall R t0;
                 (   t0 >= 0
                  -> \forall R ts0;  (0 <= ts0 & ts0 <= t0 -> -b * ts0 + v >= 0 & ts0 + 0 <= ep)
                  ->      2 * b * (m - 1 / 2 * (-b * t0 ^ 2 + 2 * t0 * v + 2 * z))
                      >= (-b * t0 + v) ^ 2
                       - d ^ 2
                    & -b * t0 + v >= 0
                    & d >= 0))
         & (      m - z
               >  (amax / b + 1) * ep * v
                + (v ^ 2 - d ^ 2) / (2 * b)
                + (amax / b + 1) * amax * ep ^ 2 / 2
            -> \forall R t2;
                 (   t2 >= 0
                  -> \forall R ts2;  (0 <= ts2 & ts2 <= t2 -> a_3 * ts2 + v >= 0 & ts2 + 0 <= ep)
                  ->      2 * b * (m - 1 / 2 * (a_3 * t2 ^ 2 + 2 * t2 * v + 2 * z))
                      >= (a_3 * t2 + v) ^ 2
                       - d ^ 2
                    & a_3 * t2 + v >= 0
                    & d >= 0)))
```

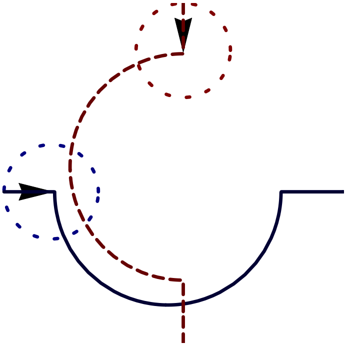
Outline Background

Air Traffic Control





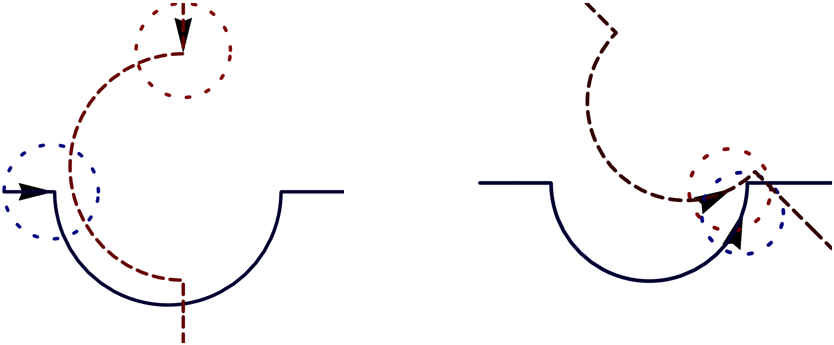
Air Traffic Control



Verification?

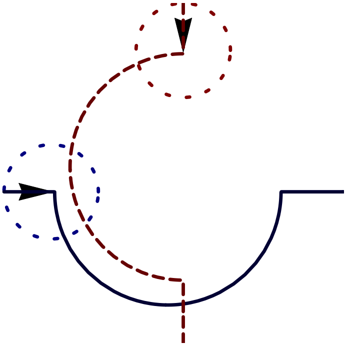
looks correct

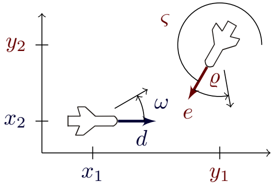
Air Traffic Control



Verification?

looks correct NO!



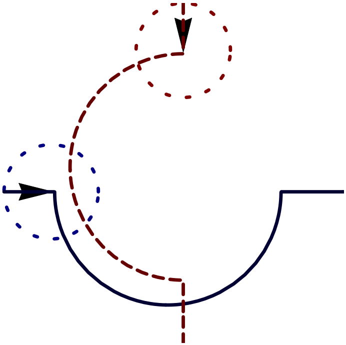


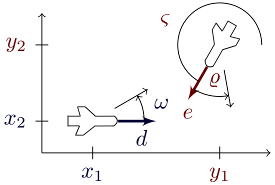
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

Verification?

looks correct NO!

Air Traffic Control



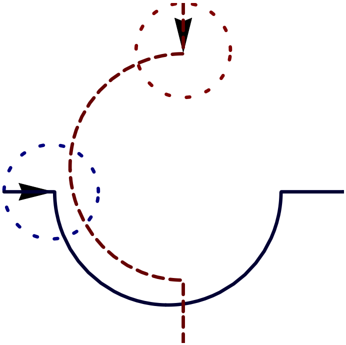


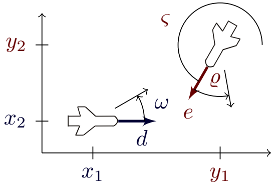
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

Example ("Solving" differential equations)

$$x_1(t) = \frac{1}{\omega\varpi}\big(x_1\omega\varpi \cos t\omega - v_2\omega \cos t\omega \sin\vartheta + v_2\omega \cos t\omega \cos t\varpi \sin\vartheta - v_1\varpi \sin t\omega$$
$$+ x_2\omega\varpi \sin t\omega - v_2\omega \cos\vartheta \cos t\varpi \sin t\omega - v_2\omega\sqrt{1 - \sin\vartheta^2} \sin t\omega$$
$$+ v_2\omega \cos\vartheta \cos t\omega \sin t\varpi + v_2\omega \sin\vartheta \sin t\omega \sin t\varpi\big) \ldots$$

Air Traffic Control





$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varpi - \omega \end{bmatrix}$$

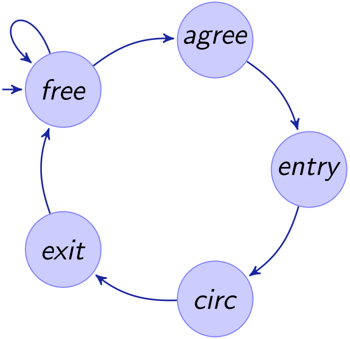
Example ("Solving" differential equations)

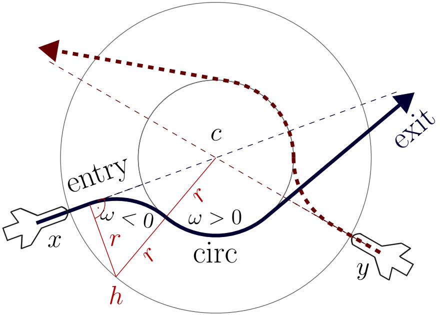
$$\forall t \geq 0 \quad \frac{1}{\omega\varpi}\big(x_1\omega\varpi \cos t\omega - v_2\omega \cos t\omega \sin\vartheta + v_2\omega \cos t\omega \cos t\varpi \sin\vartheta - v_1\varpi \sin t\omega$$

$$+ x_2\omega\varpi \sin t\omega - v_2\omega \cos\vartheta \cos t\varpi \sin t\omega - v_2\omega\sqrt{1 - \sin\vartheta^2} \sin t\omega$$

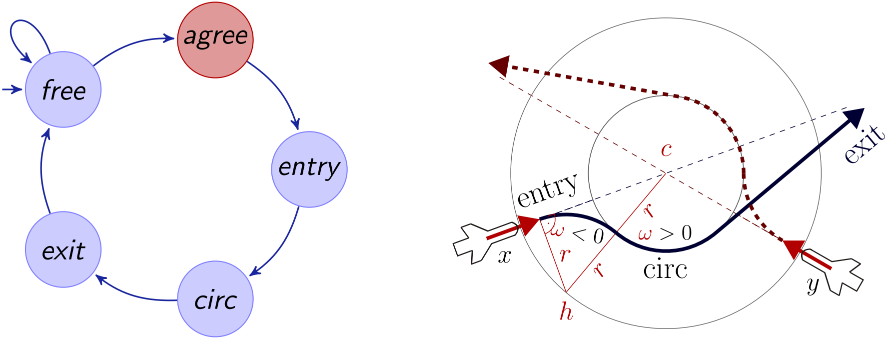
$$+ v_2\omega \cos\vartheta \cos t\omega \sin t\varpi + v_2\omega \sin\vartheta \sin t\omega \sin t\varpi\big) \ldots$$

Flyable Roundabout Maneuver: Overview

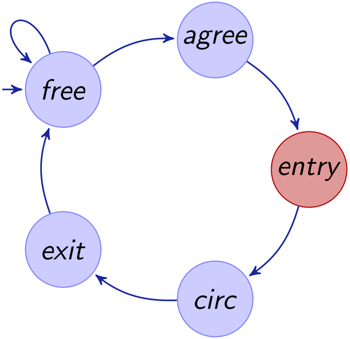


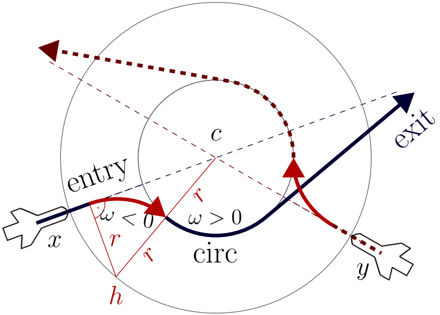


Flyable Roundabout Maneuver: Overview

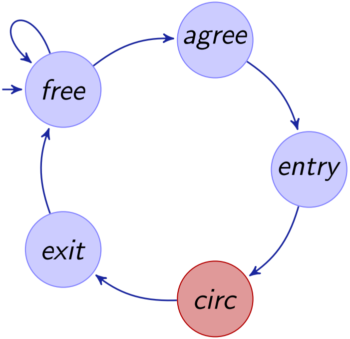


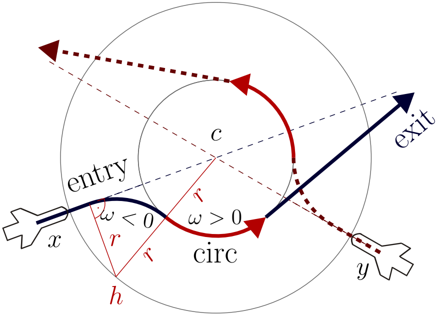
Flyable Roundabout Maneuver: Overview



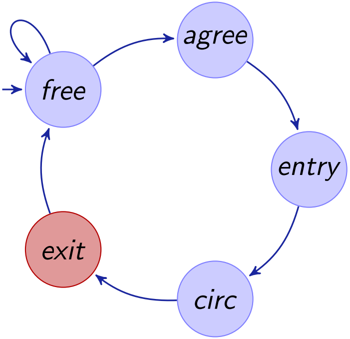


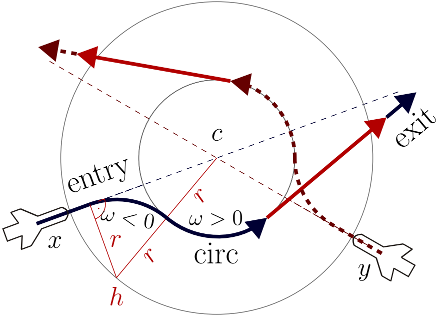
Flyable Roundabout Maneuver: Overview



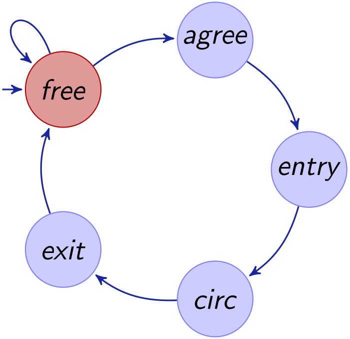


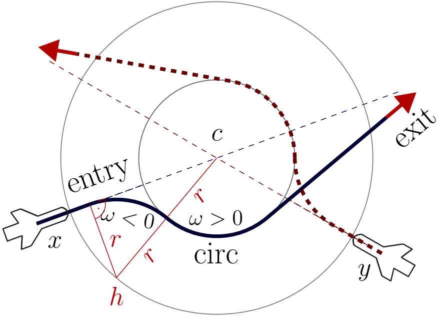
Flyable Roundabout Maneuver: Overview

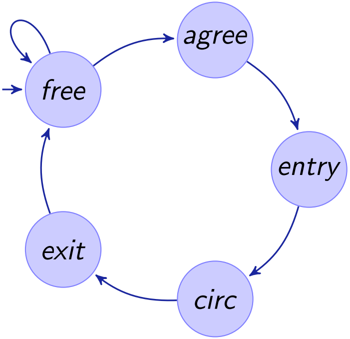


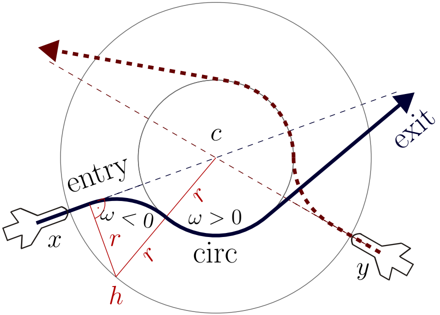


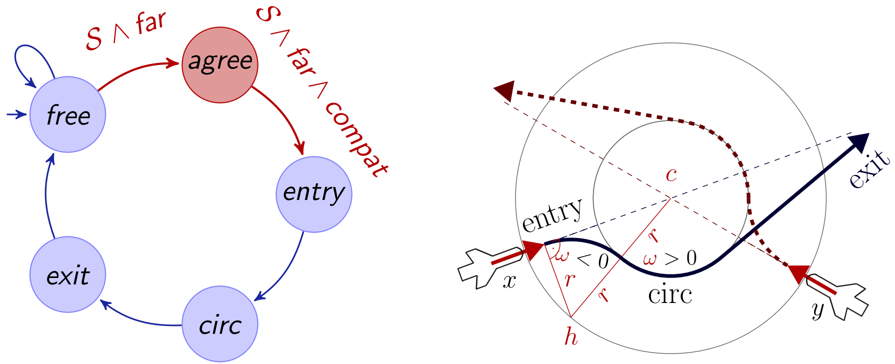
Flyable Roundabout Maneuver: Overview





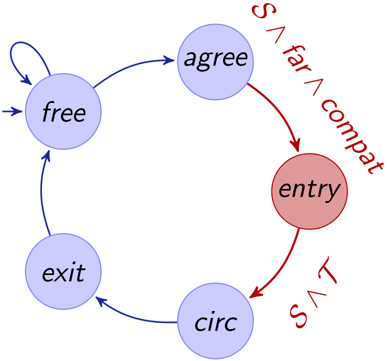

agree
free
entry
exit
circ

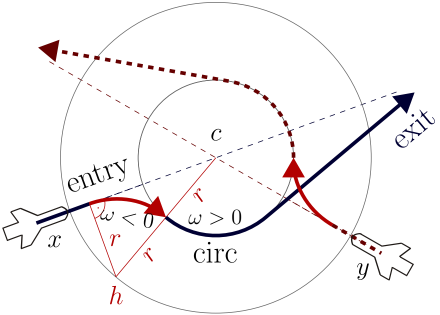

c
entry
exit
$\omega < 0$
$\omega > 0$
r
r
r
x
y
circ
h



Example (d$\mathcal{L}$ formula of verification subgoal)

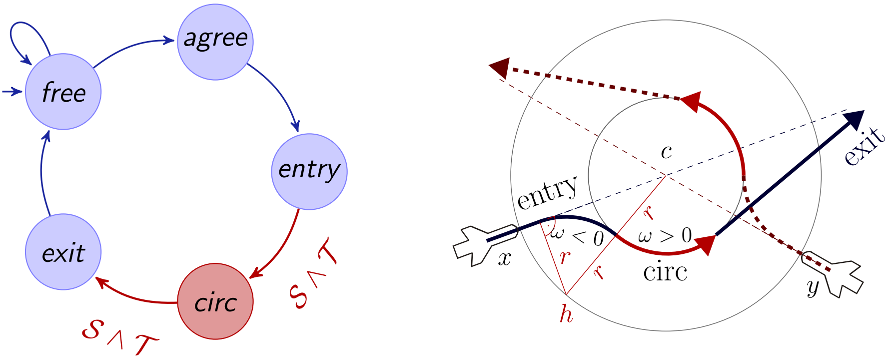
$$\mathit{safe} \land \mathit{far} \rightarrow [\mathit{agree}](\mathit{safe} \land \mathit{far} \land \mathit{compatible})$$





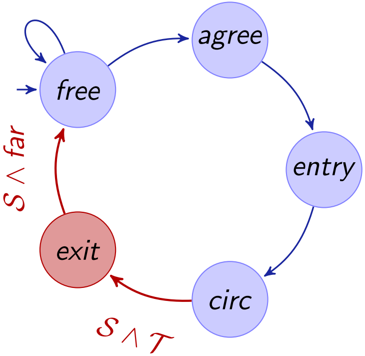
Example (d$\mathcal{L}$ formula of verification subgoal)

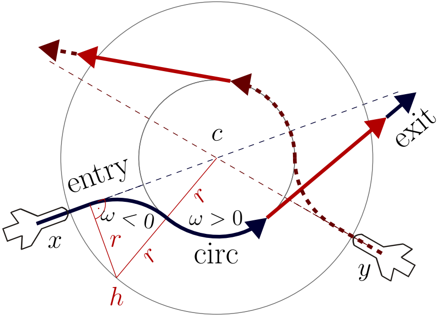
$$safe \land far \land compatible \rightarrow [entry](safe \land tangential)$$



Example (d$\mathcal{L}$ formula of verification subgoal)

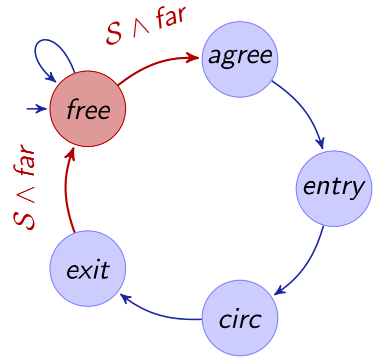
$$\mathit{safe} \wedge \mathit{tangential} \rightarrow [\mathit{circ}](\mathit{safe} \wedge \mathit{tangential})$$

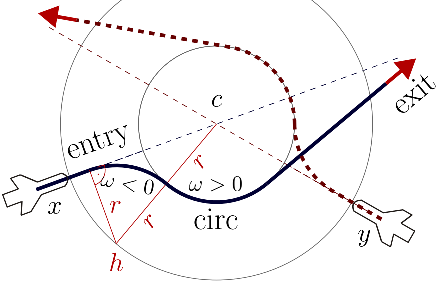




Example (d$\mathcal{L}$ formula of verification subgoal)

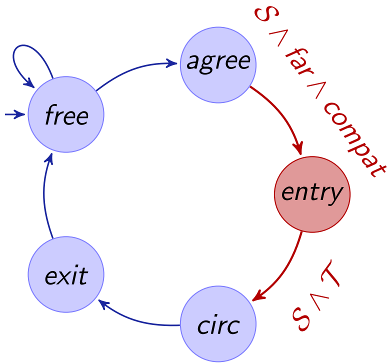
$$safe \land tangential \rightarrow [exit](safe \land far)$$

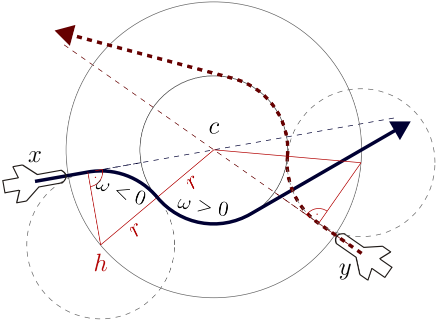




Example (d$\mathcal{L}$ formula of verification subgoal)

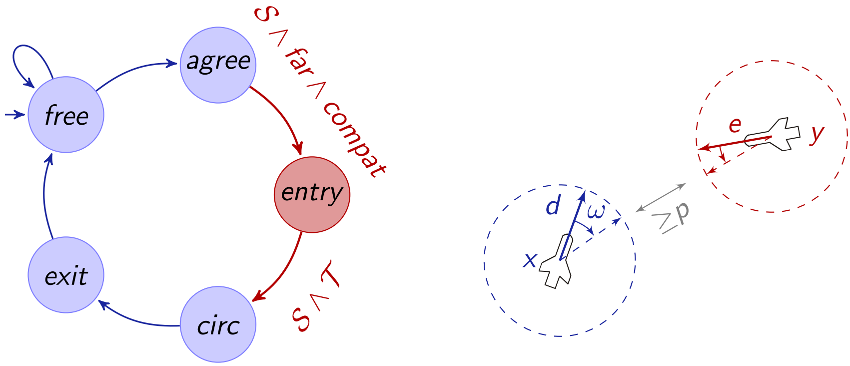
$$safe \wedge far \rightarrow [free](safe \wedge far)$$





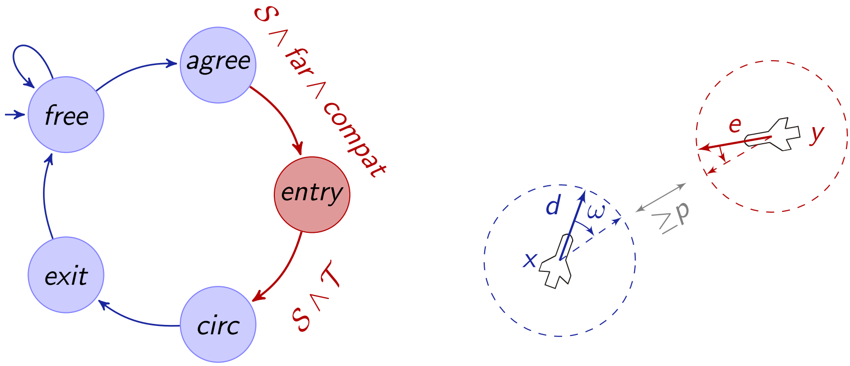
Example (d$\mathcal{L}$ formula of verification subgoal)

$$(r\omega)^2 = \|d\|^2 \land \|x - c\| = \sqrt{3}r \land \exists \lambda {\geq} 0\,(x + \lambda d = c) \land$$
$$\|h - c\| = 2r \land d = -\omega(x - h)^\perp$$
$$\to [\mathcal{F}(-\omega)\,\&\,\|x - c\| \geq r]\,\big(\|x - c\| \leq r \to d = \omega(x - c)^\perp\big)$$



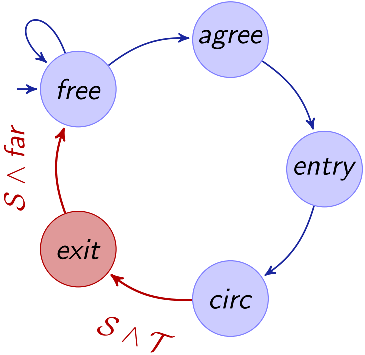
Example (d$\mathcal{L}$ formula of verification subgoal)

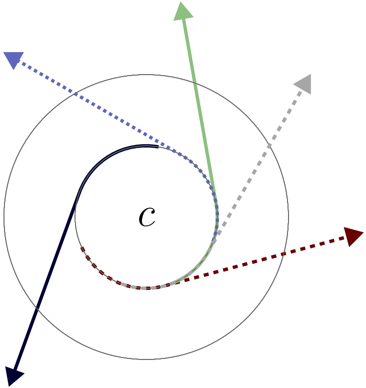
$$\|x-y\| \geq \sqrt{2}(p+2bT) \land p \geq 0 \land \|d\|^2 \leq \|e\|^2 \leq b^2 \land b \geq 0 \land T \geq 0 \rightarrow [entry]\,(\|x-y\| \geq p)$$



Example (dℒ formula of verification subgoal)

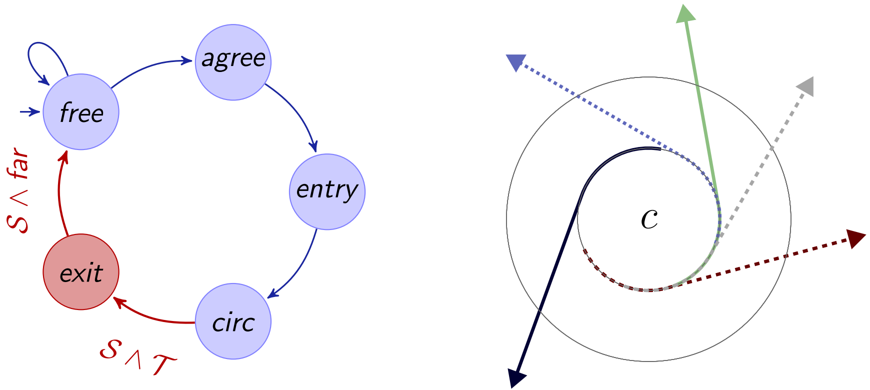
$$x = z \land \|d\|^2 \leq b^2 \land b \geq 0$$
$$\rightarrow [\tau := 0;\ \exists \omega\, \mathcal{F}(\omega) \land \tau' = 1]\,(\|x - z\|_\infty \leq \tau b)$$





Example (d$\mathcal{L}$ formula of verification subgoal)

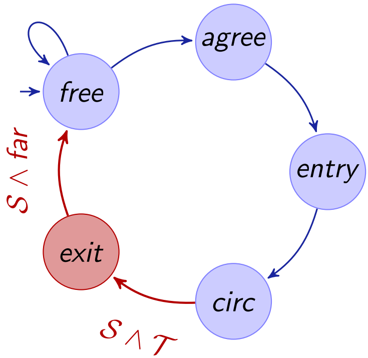
$$\mathcal{T} \wedge \|x-y\|^2 \geq p^2 \rightarrow [x' = d \wedge y' = e]\,(\|x-y\|^2 \geq p^2)$$

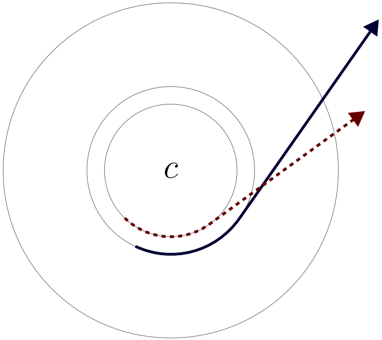


Example (d$\mathcal{L}$ formula of verification subgoal)

$$\mathcal{T} \wedge \|x - y\|^2 \geq p^2 \rightarrow [x' = d;\ y' = e]\,(\|x - y\|^2 \geq p^2)$$

Flyable Roundabout Maneuver: Exit

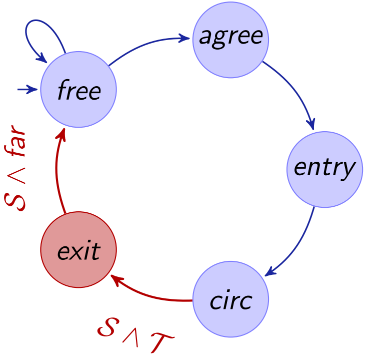


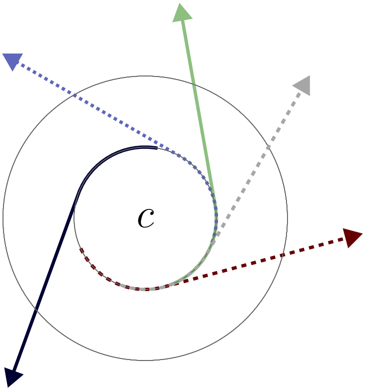


Example (d$\mathcal{L}$ formula of verification subgoal)

$$\mathcal{T} \wedge \|x - y\|^2 \geq p^2 \rightarrow [x' = d;\ y' = e](\|x - y\|^2 \geq p^2)$$

Flyable Roundabout Maneuver: Exit

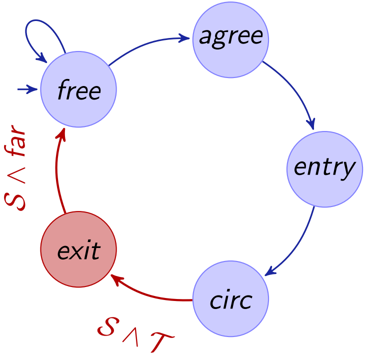


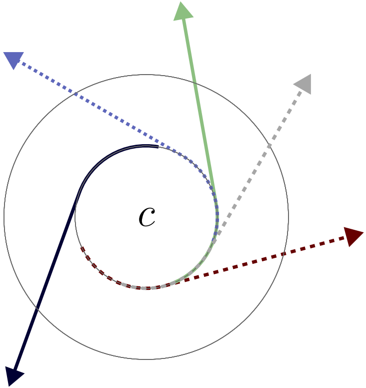


Example (d$\mathcal{L}$ formula of verification subgoal)

$$\mathcal{T} \land \|x - y\|^2 \geq p^2 \rightarrow [x' = d;\ y' = e]\,(\|x - y\|^2 \geq p^2)$$

Flyable Roundabout Maneuver: Exit





Example (d$\mathcal{L}$ formula of verification subgoal)

$$\mathcal{T} \wedge d \neq e \rightarrow \forall a \, \langle x' = d \wedge y' = e \rangle \, (\|x - y\|^2 > a^2)$$

provable automatically!

$$
\begin{aligned}
\psi &\equiv \phi \to [trm^*]\phi \\
\phi &\equiv \|x - y\|^2 \geq p^2 \equiv (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2 \\
trm &\equiv free;\quad entry;\quad \mathcal{F}(\omega) \land \mathcal{G}(\omega) \\
free &\equiv \exists \omega\, \mathcal{F}(\omega) \land \exists \varpi\, \mathcal{G}(\varpi) \land \phi \\
entry &\equiv \exists u\, \omega := u;\ \exists c\, (d := \omega(x - c)^{\perp} \land e := \omega(y - c)^{\perp})
\end{aligned}
$$

$$
\mathcal{F}(\omega) \equiv \begin{pmatrix} x_1' = v\cos\vartheta & = d_1 \\ \land\, x_2' = v\sin\vartheta & = d_2 \\ \land\, d_1' = v(-\sin\vartheta)\vartheta' & = -\omega d_2 \\ \land\, d_2' = v(\cos\vartheta)\vartheta' & = \omega d_1 \end{pmatrix} \quad \mathcal{G}(\varpi) \equiv \begin{pmatrix} y_1' = e_1 \\ \land\, y_2' = e_2 \\ \land\, e_1' = -\varpi e_2 \\ \land\, e_2' = \varpi e_1 \end{pmatrix}
$$

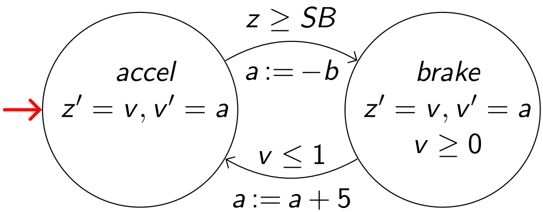
provable automatically!

$$
\begin{aligned}
\psi &\equiv \phi \to [trm^*]\phi \\
\phi &\equiv (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2 \land (y_1 - z_1)^2 + (y_2 - z_2)^2 \geq p^2 \\
&\quad \land (x_1 - z_1)^2 + (x_2 - z_2)^2 \geq p^2 \land (x_1 - u_1)^2 + (x_2 - u_2)^2 \geq p^2 \\
&\quad \land (y_1 - u_1)^2 + (y_2 - u_2)^2 \geq p^2 \land (z_1 - u_1)^2 + (z_2 - u_2)^2 \geq p^2 \\
trm &\equiv free;\ \ entry; \\
&\quad x_1' = d_1 \land x_2' = d_2 \land d_1' = -\omega_x d_2 \land d_2' = \omega_x d_1 \\
&\quad \land y_1' = e_1 \land y_2' = e_2 \land e_1' = -\omega_y e_2 \land e_2' = \omega_y e_1 \\
&\quad \land z_1' = f_1 \land z_2' = f_2 \land f_1' = -\omega_z f_2 \land f_2' = \omega_z f_1 \\
&\quad \land u_1' = g_1 \land u_2' = g_2 \land g_1' = -\omega_u g_2 \land g_2' = \omega_u g_1 \\
free &\equiv (\omega_x := *;\ \ \omega_y := *;\ \ \omega_z := *;\ \ \omega_u := *; \\
&\quad x_1' = d_1 \land x_2' = d_2 \land d_1' = -\omega_x d_2 \land d_2' = \omega_x d_1 \\
&\quad \land y_1' = e_1 \land y_2' = e_2 \land e_1' = -\omega_y e_2 \land e_2' = \omega_y e_1 \\
&\quad \land z_1' = f_1 \land z_2' = f_2 \land f_1' = -\omega_z f_2 \land f_2' = \omega_z f_1 \\
&\quad \land u_1' = g_1 \land u_2' = g_2 \land g_1' = -\omega_u g_2 \land g_2' = \omega_u g_1 \land \phi)^* \\
entry &\equiv \omega := *;\ c := *; \\
&\quad d_1 := -\omega(x_2 - c_2);\ d_2 := \omega(x_1 - c_1); \\
&\quad e_1 := -\omega(y_1 - c_1);\ e_2 := \omega(y_2 - c_2); \\
&\quad f_1 := -\omega(z_1 - c_1);\ f_2 := \omega(z_2 - c_2); \\
&\quad g_1 := -\omega(u_1 - c_1);\ g_2 := \omega(u_2 - c_2)
\end{aligned}
$$

Experimental Results (Air Traffic Control)

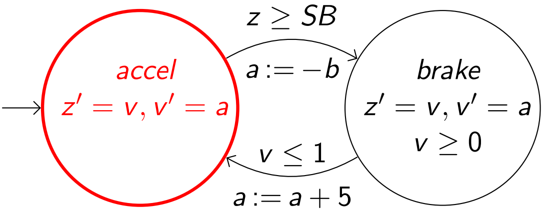
Case Study	Time(s)	Mem(Mb)	Steps	Dim
tangential roundabout (2a/c)	10.4	6.8	197	13
tangential roundabout (3a/c)	253.6	7.2	342	18
tangential roundabout (4a/c)	382.9	10.2	520	23
tangential roundabout (5a/c)	1882.9	39.1	735	28
bounded maneuver speed	0.5	6.3	14	4
flyable roundabout entry*	10.1	9.6	132	8
flyable entry feasible*	104.5	87.9	16	10
flyable entry circular	3.2	7.6	81	5
limited entry progress	1.9	6.5	60	8
entry separation	140.1	20.1	512	16
mutual negotiation successful	0.8	6.4	60	12
mutual negotiation feasible*	7.5	23.8	21	11
mutual far negotiation	2.4	8.1	67	14
simultaneous exit separation*	4.3	12.9	44	9
different exit directions	3.1	11.1	42	11

Outline Background



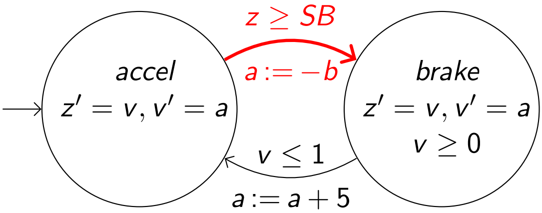
$\rightsquigarrow$

$$
\begin{aligned}
& q := accel; \\
& (\quad (?q = accel; \quad z' = v, v' = a) \\
& \cup \ (?q = accel \land z \geq SB; \quad a := -b; \quad q := brake; \quad ?v \geq 0) \\
& \cup \ (?q = brake; \quad z' = v, v' = a \,\&\, v \geq 0) \\
& \cup \ (?q = brake \land v \leq 1; \quad a := a + 5; \quad q := accel))^*
\end{aligned}
$$



$\rightsquigarrow$

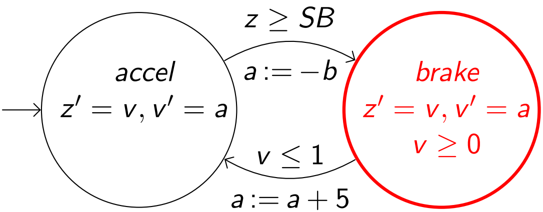
$$
\begin{aligned}
&q := accel;\\
&(\quad (?q = accel;\quad z' = v, v' = a)\\
&\cup\ (?q = accel \land z \geq SB;\quad a := -b;\quad q := brake;\quad ?v \geq 0)\\
&\cup\ (?q = brake;\quad z' = v, v' = a \,\&\, v \geq 0)\\
&\cup\ (?q = brake \land v \leq 1;\quad a := a + 5;\quad q := accel)\big)^*
\end{aligned}
$$



$\rightsquigarrow$

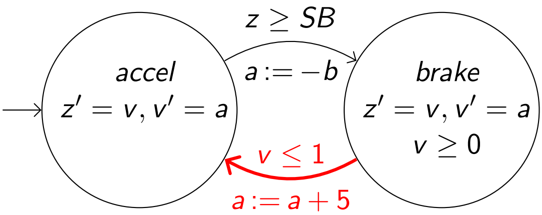
$$
\begin{aligned}
&q := accel;\\
&(\quad (?q = accel;\quad z' = v, v' = a)\\
&\cup\ (?q = accel \land z \geq SB;\quad a := -b;\quad q := brake;\quad ?v \geq 0)\\
&\cup\ (?q = brake;\quad z' = v, v' = a \,\&\, v \geq 0)\\
&\cup\ (?q = brake \land v \leq 1;\quad a := a + 5;\quad q := accel))^*
\end{aligned}
$$

Embedding Hybrid Automata as Hybrid Programs



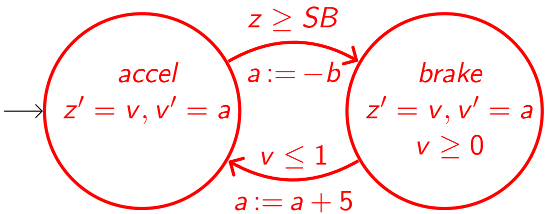
$\rightsquigarrow$

$$
\begin{aligned}
& q := accel; \\
& \big(\quad (?q = accel; \quad z' = v, v' = a) \\
& \cup \; (?q = accel \land z \geq SB; \quad a := -b; \quad q := brake; \quad ?v \geq 0) \\
& \cup \; (?q = brake; \quad z' = v, v' = a \,\&\, v \geq 0) \\
& \cup \; (?q = brake \land v \leq 1; \quad a := a + 5; \quad q := accel) \big)^*
\end{aligned}
$$



$\rightsquigarrow$

$$
\begin{aligned}
&q := accel;\\
&(\quad (?q = accel;\quad z' = v, v' = a)\\
&\cup\ (?q = accel \land z \geq SB;\quad a := -b;\quad q := brake;\quad ?v \geq 0)\\
&\cup\ (?q = brake;\quad z' = v, v' = a \,\&\, v \geq 0)\\
&\cup\ (?q = brake \land v \leq 1;\quad a := a + 5;\quad q := accel))^*
\end{aligned}
$$



$$\rightsquigarrow$$

$$
\begin{aligned}
& q := accel; \\
& \big(\quad (?q = accel; \quad z' = v, v' = a) \\
& \cup \; (?q = accel \land z \geq SB; \quad a := -b; \quad q := brake; \quad ?v \geq 0) \\
& \cup \; (?q = brake; \quad z' = v, v' = a \,\&\, v \geq 0) \\
& \cup \; (?q = brake \land v \leq 1; \quad a := a + 5; \quad q := accel)\big)^*
\end{aligned}
$$

Outline Background

Cyber-Physical Systems:

Q: I want to verify my car

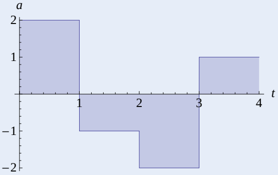
Challenge

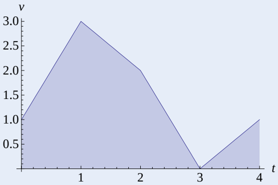
Cyber-Physical Systems: Hybrid Systems

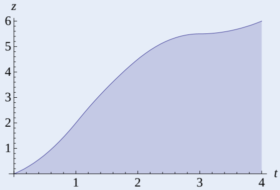
Q: I want to verify my car A: Hybrid systems

Challenge (Hybrid Systems)

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



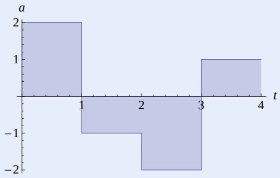


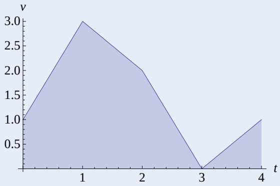


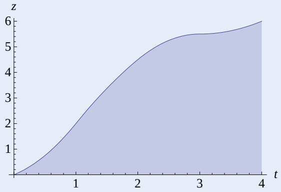
Q: I want to verify my car A: Hybrid systems Q: But there's a lot of cars!

Challenge (Hybrid Systems)

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

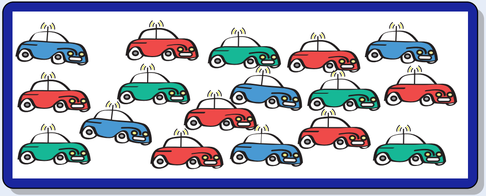






Q: I want to verify a lot of cars

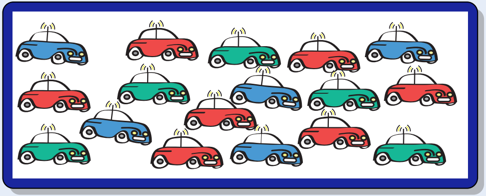
Challenge

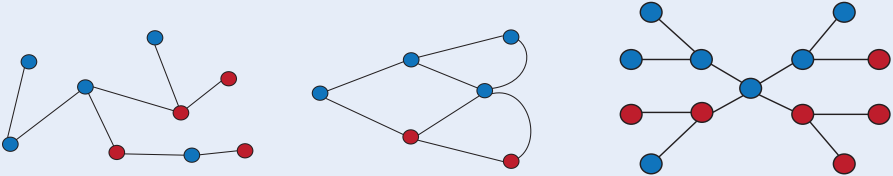


Q: I want to verify a lot of cars A: Distributed systems

Challenge (Distributed Systems)

- Local computation (finite state automaton)
- Remote communication (network graph)

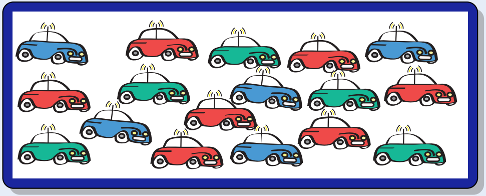


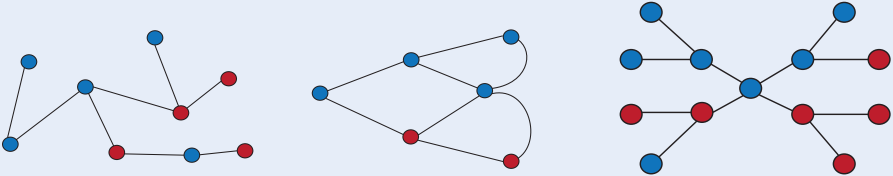


Q: I want to verify a lot of cars A: Distributed systems Q: But they move!

Challenge (Distributed Systems)

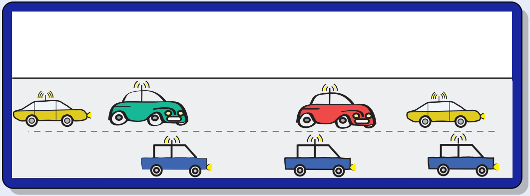
- Local computation (finite state automaton)
- Remote communication (network graph)





Q: I want to verify lots of moving cars

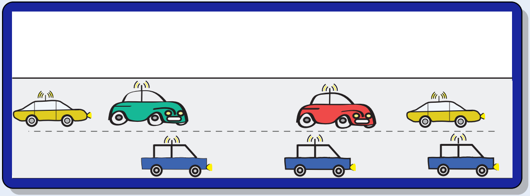
Challenge



Q: I want to verify lots of moving cars A: Distributed hybrid systems

Challenge (Distributed Hybrid Systems)

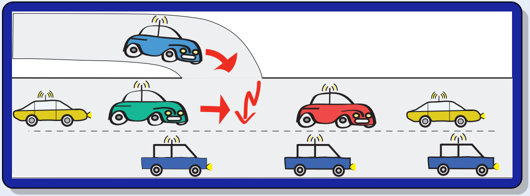
- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)
- Structural dynamics (communication/coupling)



Q: I want to verify lots of moving cars A: Distributed hybrid systems

Challenge (Distributed Hybrid Systems)

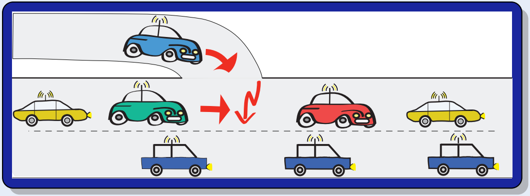
- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)
- Structural dynamics (communication/coupling)
- Dimensional dynamics (appearance)



Q: I want to verify lots of moving cars A: Distributed hybrid systems Q: How?

Challenge (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)
- Structural dynamics (communication/coupling)
- Dimensional dynamics (appearance)

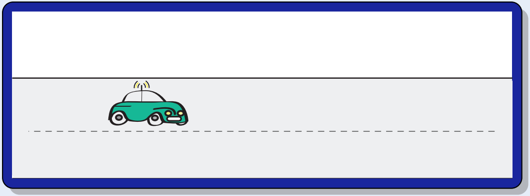


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)
- Structural dynamics (communication/coupling)

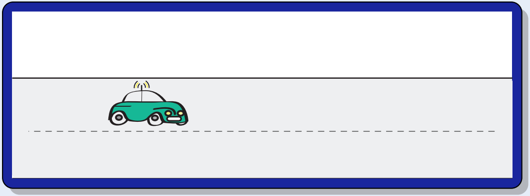


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $x'' = a$
- Discrete dynamics (control decisions)
- Structural dynamics (communication/coupling)

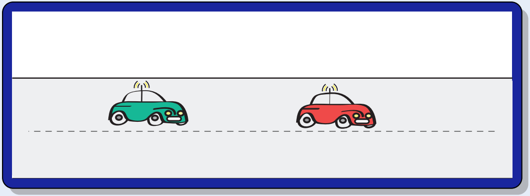


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $x'' = a$
- Discrete dynamics (control decisions)
  $a := \texttt{if} \,..\, \texttt{then}\; a\; \texttt{else}\; -b\; \texttt{fi}$
- Structural dynamics (communication/coupling)

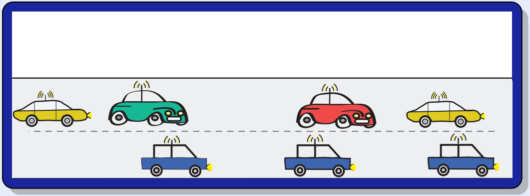


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

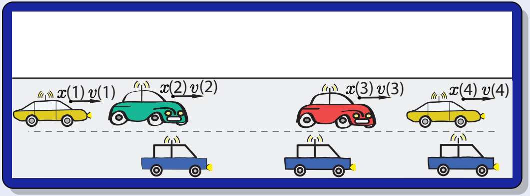
- Continuous dynamics (differential equations)
  $$x'' = a$$
- Discrete dynamics (control decisions)
  $$a := \texttt{if} \ldots \texttt{then}\, a\, \texttt{else}\, -b\, \texttt{fi}$$
- Structural dynamics (communication/coupling)



Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $$x'' = a$$
- Discrete dynamics (control decisions)
  $$a := \texttt{if} \,..\, \texttt{then}\, a \,\texttt{else}\, -b\, \texttt{fi}$$
- Structural dynamics (communication/coupling)



Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

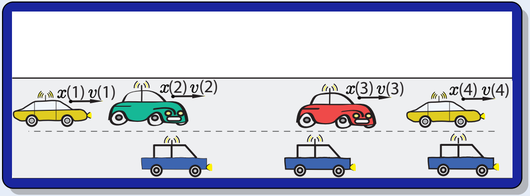
- Continuous dynamics (differential equations)

  $$x(i)'' = a(i)$$

- Discrete dynamics (control decisions)

  $$a(i) := \texttt{if} .. \texttt{then}\ a\ \texttt{else}\ -b\ \texttt{fi}$$

- Structural dynamics (communication/coupling)



Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

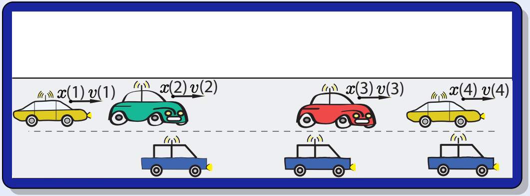
- Continuous dynamics (differential equations)

  $\forall i\; x(i)'' = a(i)$

- Discrete dynamics (control decisions)

$\forall i\; a(i) := \texttt{if} \,..\, \texttt{then}\; a \;\texttt{else}\; -b \;\texttt{fi}$

- Structural dynamics (communication/coupling)

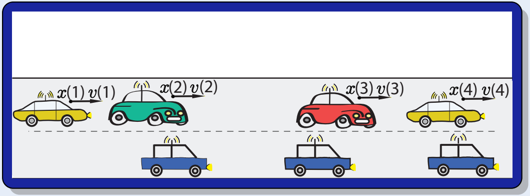


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $\forall i\; x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\; a(i) := \texttt{if} .. \texttt{then}\; a\; \texttt{else}\; -b\; \texttt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := \mathit{carInFrontOf}(i)$

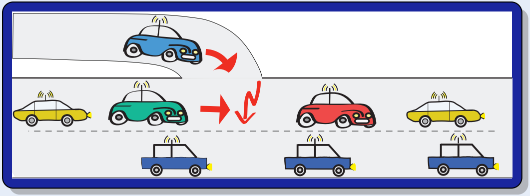


Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)

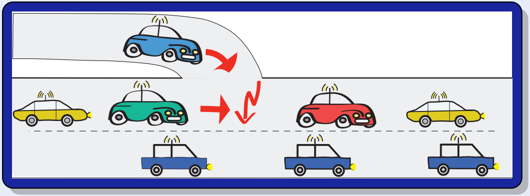
- Continuous dynamics (differential equations)
  $\forall i\ x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\ a(i) := \texttt{if .. then}\ a\ \texttt{else}\ -b\ \texttt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := carInFrontOf(i)$
- Dimensional dynamics (appearance)



Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

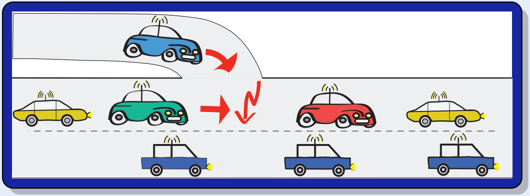
Model (Distributed Hybrid Systems)



- Continuous dynamics (differential equations)
  $\forall i\, x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\, a(i) := \mathtt{if}\ ..\ \mathtt{then}\ a\ \mathtt{else}\ -b\ \mathtt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := \mathit{carInFrontOf}(i)$
- Dimensional dynamics (appearance)
  $n := \mathtt{new}\ \mathit{Car}$

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)



- Continuous dynamics (differential equations)
  $\forall i\, x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\, a(i) := \mathtt{if} \,..\, \mathtt{then}\ a\ \mathtt{else}\ -b\ \mathtt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := carInFrontOf(i)$
- Dimensional dynamics (appearance)
  $n := \mathtt{new}\ Car$

$\Rightarrow$ Communication
$d(i, \ell(i)) := d(i, \ell(i)) + 10$

Model for Distributed Hybrid Systems

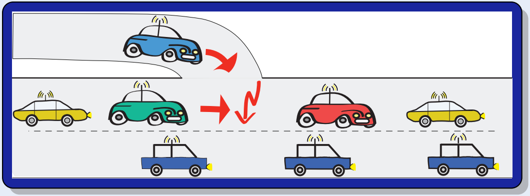
Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $\forall i\, x(i)'' = a(i)$
- Discrete dynamics (control decisions)

$\forall i\, a(i) := \mathtt{if}\, ..\, \mathtt{then}\, a\, \mathtt{else}\, -b\, \mathtt{fi}$

- Structural dynamics (communication/coupling)
  $\ell(i) := carInFrontOf(i)$
- Dimensional dynamics (appearance)
  $n := \mathtt{new}\ Car$

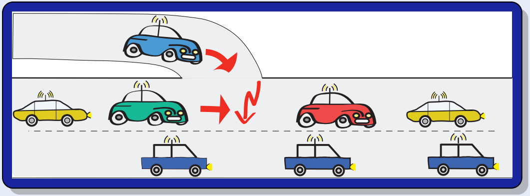


$\Rightarrow$ Communication

$\forall i\, d(i, \ell(i)) := d(i, \ell(i)) + 10$

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)



- Continuous dynamics (differential equations)

  $\forall i\; x(i)'' = a(i)$

- Discrete dynamics (control decisions)

$\forall i\; a(i) := \texttt{if .. then}\; a\; \texttt{else}\; -b\; \texttt{fi}$

- Structural dynamics (communication/coupling)

  $\ell(i) := carInFrontOf(i)$

- Dimensional dynamics (appearance)

  $n := \texttt{new}\; Car$

$\Rightarrow$ Communication

$\forall i\; d(i, \ell(i)) := d(i, \ell(i)) + 10$

$\Rightarrow$ Discrete structural dynamics

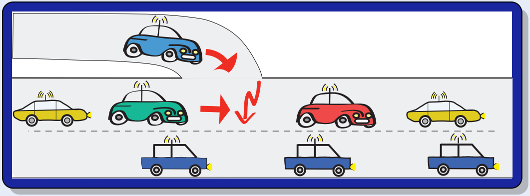
$\ell(i) := \ell(\ell(i))$

Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)

- Continuous dynamics (differential equations)
  $\forall i\, x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\, a(i) := \texttt{if} .. \texttt{then}\, a\, \texttt{else} -b\, \texttt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := carInFrontOf(i)$
- Dimensional dynamics (appearance)
  $n := \texttt{new}\ Car$



$\Rightarrow$ Communication
$\forall i\, d(i, \ell(i)) := d(i, \ell(i)) + 10$

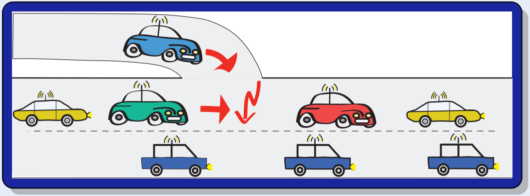
$\Rightarrow$ Discrete structural dynamics
$\ell(i) := \ell(\ell(i))$

$\Rightarrow$ Continuous structural dynamics
$x(i)'' = a(i) + c(i, \ell(i))a(\ell(i))$

Model for Distributed Hybrid Systems

Q: How to model distributed hybrid systems A: Quantified Hybrid Programs

Model (Distributed Hybrid Systems)



- Continuous dynamics (differential equations)
  $\forall i\, x(i)'' = a(i)$
- Discrete dynamics (control decisions)
  $\forall i\, a(i) := \mathtt{if} .. \mathtt{then}\, a\, \mathtt{else}\, -b\, \mathtt{fi}$
- Structural dynamics (communication/coupling)
  $\ell(i) := carInFrontOf(i)$
- Dimensional dynamics (appearance)
  $n := \mathtt{new}\ Car$

$\Rightarrow$ Communication
$\forall i\, d(i, \ell(i)) := d(i, \ell(i)) + 10$

$\Rightarrow$ Discrete structural dynamics
$\ell(i) := \ell(\ell(i))$

$\Rightarrow$ Continuous structural dynamics
$\forall i\, x(i)'' = a(i) + c(i, \ell(i))a(\ell(i))$

Quantified Differential Dynamic Logic Qd$\mathcal{L}$: Syntax

Definition (Quantified hybrid program α)

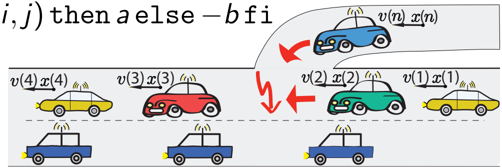
$\forall i : C \; x(i)' = \theta$	(quantified ODE)	
$\forall i : C \; x(i) := \theta$	(quantified assignment)	jump & test
$?\chi$	(conditional execution)	
$\alpha ; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

Definition (Quantified hybrid program α)

$\forall i : C\ x(i)' = \theta$	(quantified ODE)	
$\forall i : C\ x(i) := \theta$	(quantified assignment)	jump & test
$?\chi$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

$$DCCS \equiv (ctrl; drive)^*$$

$$ctrl \equiv \forall i : C\ a(i) := \mathtt{if}\ \forall j : C\ far(i,j)\ \mathtt{then}\ a\ \mathtt{else}\ -b\ \mathtt{fi}$$
$$drive \equiv \forall i : C\ x(i)'' = a(i)$$



Quantified Differential Dynamic Logic Qd$\mathcal{L}$: Syntax

Definition (Quantified hybrid program α)

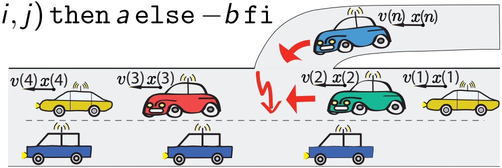
$\forall i : C\ x(i)' = \theta$	(quantified ODE)	
$\forall i : C\ x(i) := \theta$	(quantified assignment)	jump & test
$?\chi$	(conditional execution)	
$\alpha ; \beta$	(seq. composition)	Kleene algebra
$\alpha \cup \beta$	(nondet. choice)	
α^*	(nondet. repetition)	

$$DCCS \equiv (appear; ctrl; drive)^*$$

$$appear \equiv n := \texttt{new}\ C;\quad ?(\forall j : C\ far(j, n))$$

$$ctrl \equiv \forall i : C\ a(i) := \texttt{if}\ \forall j : C\ far(i, j)\ \texttt{then}\ a\ \texttt{else}\ {-b}\ \texttt{fi}$$

$$drive \equiv \forall i : C\ x(i)'' = a(i)$$



Quantified Differential Dynamic Logic Qd$\mathcal{L}$: Syntax

Definition (Quantified hybrid program α)

$\forall i : C\ x(i)' = \theta$	(quantified ODE)	
$\forall i : C\ x(i) := \theta$	(quantified assignment)	jump & test
$?\chi$	(conditional execution)	
$\alpha; \beta$	(seq. composition)	
$\alpha \cup \beta$	(nondet. choice)	Kleene algebra
α^*	(nondet. repetition)	

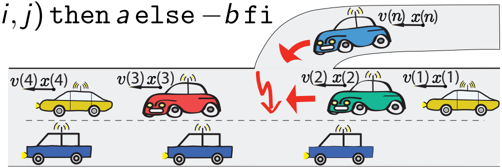
$$DCCS \equiv (appear\, ;\, ctrl\, ;\, drive)^*$$

$$appear \equiv n := \mathtt{new}\ C;\quad ?(\forall j : C\ far(j, n))$$

$$ctrl \equiv \forall i : C\ a(i) := \mathtt{if}\ \forall j : C\ far(i, j)\ \mathtt{then}\ a\ \mathtt{else}\ -b\ \mathtt{fi}$$

$$drive \equiv \forall i : C\ x(i)'' = a(i)$$

new C is definable!



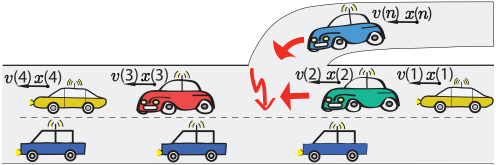
Definition (QdL Formula ϕ)

$\neg, \wedge, \vee, \rightarrow, \ \forall x, \exists x, \ =, \leq, \ +, \cdot$ (ℝ-first-order part)

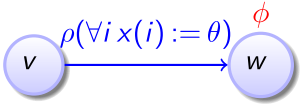
$[\alpha]\phi, \quad \langle\alpha\rangle\phi$ (dynamic part)

$$\forall i, j : C \ far(i,j) \rightarrow [(appear; ctrl; drive)^*] \ \forall i {\neq} j : C \ x(i) \neq x(j)$$

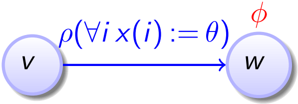
$$far(i,j) \equiv i \neq j \rightarrow x(i) < x(j) \wedge v(i) \leq v(j) \wedge a(i) \leq a(j) \\ \vee x(i) > x(j) \wedge v(i) \geq v(j) \wedge a(i) \geq a(j) \ldots$$



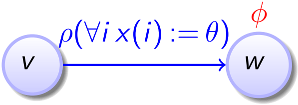
$$\frac{\forall i\,(i = u \to \phi(\theta))}{\phi([\forall i\, x(i) := \theta]x(u))}$$



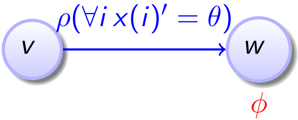
$$\frac{\forall i\,(i = [\forall i\, x(i) := \theta]u \to \phi(\theta))}{\phi([\forall i\, x(i) := \theta]x(u))}$$



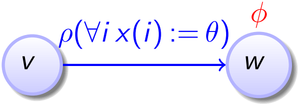
$$\frac{\forall i\,(i = [\forall i\, x(i) := \theta]u \rightarrow \phi(\theta))}{\phi([\forall i\, x(i) := \theta]x(u))}$$



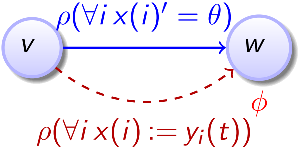
$$\frac{\exists t{\geq}0\, \langle \forall i\, x(i) := y_i(t)\rangle \phi}{\langle \forall i\, x(i)' = \theta\rangle \phi}$$



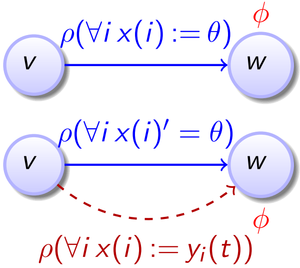
$$\frac{\forall i\,(i = [\forall i\, x(i) := \theta]u \rightarrow \phi(\theta))}{\phi([\forall i\, x(i) := \theta]x(u))}$$



$$\frac{\exists t{\geq}0\, \langle \forall i\, x(i) := y_i(t)\rangle \phi}{\langle \forall i\, x(i)' = \theta\rangle \phi}$$



$$\frac{\forall i\, (i = [\forall i\, x(i) := \theta]u \to \phi(\theta))}{\phi([\forall i\, x(i) := \theta]x(u))}$$

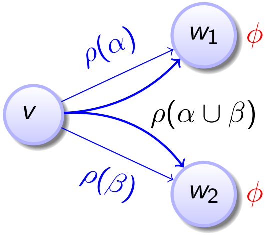


$$\frac{\exists t{\geq}0\, \langle \forall i\, x(i) := y_i(t)\rangle \phi}{\langle \forall i\, x(i)' = \theta\rangle \phi}$$

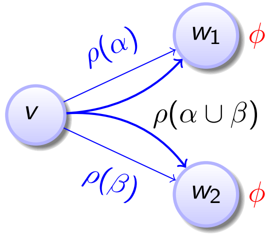
solve infinite-dimensional diff. eqn.?

compositional semantics $\Rightarrow$ compositional rules!

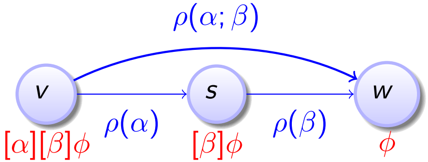
$$\frac{[\alpha]\phi \land [\beta]\phi}{[\alpha \cup \beta]\phi}$$



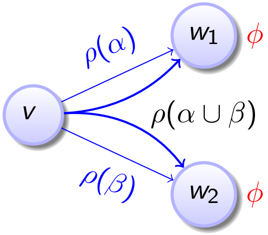
$$\frac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi}$$



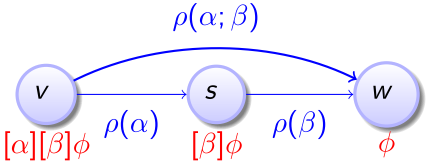
$$\frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi}$$



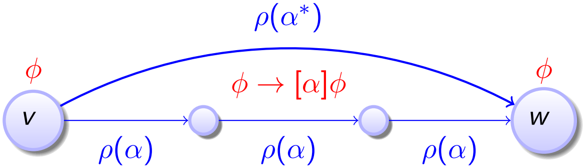
$$\frac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi}$$



$$\frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi}$$



$$\frac{\phi \quad (\phi \to [\alpha]\phi)}{[\alpha^*]\phi}$$



Soundness and Completeness

Theorem (Relative Completeness)

Qd$\mathcal{L}$ *calculus is a sound & complete axiomatisation of distributed hybrid systems relative to quantified differential equations.* ▸ Proof 16p.

André Platzer.
Quantified differential dynamic logic for distributed hybrid systems.
In Anuj Dawar and Helmut Veith, editors,
CSL, vol. 6247 of *LNCS*, 469–483. Springer, 2010.

Soundness and Completeness

Theorem (Relative Completeness)

Qd$\mathcal{L}$ *calculus is a sound & complete axiomatisation of distributed hybrid systems relative to quantified differential equations.*

Proof 16p.

Corollary (Proof-theoretical Alignment)

proving distributed hybrid systems = proving dynamical systems!

André Platzer.
Quantified differential dynamic logic for distributed hybrid systems.
In Anuj Dawar and Helmut Veith, editors,
CSL, vol. 6247 of *LNCS*, 469–483. Springer, 2010.

Soundness and Completeness

Theorem (Relative Completeness)

QdL calculus is a sound & complete axiomatisation of distributed hybrid systems relative to quantified differential equations.

Proof 16p.

Corollary (Proof-theoretical Alignment)

proving distributed hybrid systems = proving dynamical systems!

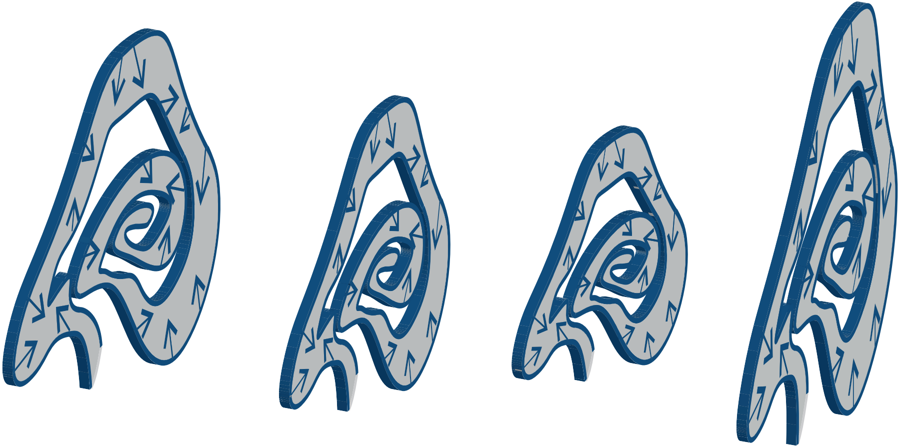
Corollary (Decomposition!)

distributed hybrid systems can be verified by recursive decomposition

André Platzer.
Quantified differential dynamic logic for distributed hybrid systems.
In Anuj Dawar and Helmut Veith, editors,
CSL, vol. 6247 of *LNCS*, 469–483. Springer, 2010.

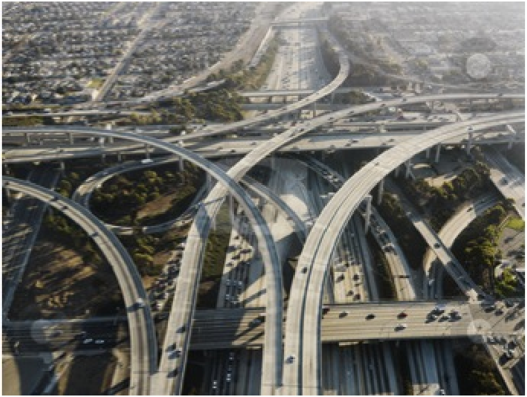
Definition (Quantified Differential Invariant)

Quantified formula F closed under total differentiation with respect to quantified differential constraints



Outline Background

Driver's License Test for Robotic Cars?



Driver's License Test for Robotic Cars?

Driver's License Test for Robotic Cars? Proof!

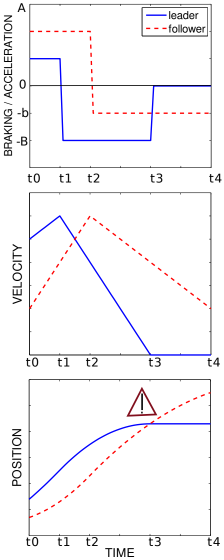
Car Control: Local Lane Control Challenge

Challenge: Local lane dynamics

- A car controller for a differential equation respects separation of local lane.

Challenge: Local lane dynamics

- A car controller for a differential equation respects separation of local lane.
- Follower car maintains safe distance to leader:

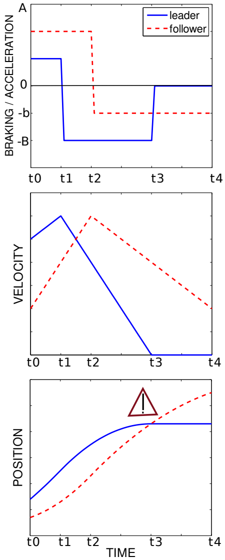


Car Control: Local Lane Control Challenge

Challenge: Local lane dynamics

- A car controller for a differential equation respects separation of local lane.
- Follower car maintains safe distance to leader:

$$f \ll \ell \rightarrow [(a_i := ctrl;\ x_i'' = a_i)^*]\, f \ll \ell$$

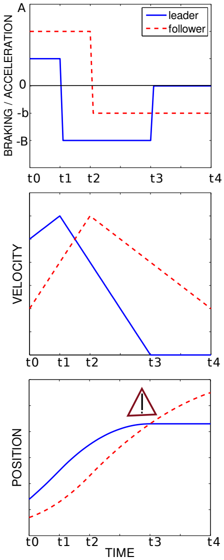


Challenge: Local lane dynamics

- A car controller for a differential equation respects separation of local lane.
- Follower car maintains safe distance to leader:

$$f \ll \ell \rightarrow [(a_i := ctrl;\ x_i'' = a_i)^*]\, f \ll \ell$$

$$f \ll \ell \equiv\ (x_f \le x_\ell) \land (f \ne \ell) \rightarrow \\ (x_\ell > x_f + \frac{v_f^2}{2b} - \frac{v_\ell^2}{2B} \\ \land x_\ell > x_f \land v_f \ge 0 \land v_\ell \ge 0)$$



$$f \ll \ell \to [\texttt{llc}]\, f \ll \ell$$

Hybrid Program (Local lane control)

$$\texttt{llc} \equiv (ctrl; dyn)^*$$

$$ctrl \equiv \ell_{ctrl} \;||\; f_{ctrl};$$

$$\ell_{ctrl} \equiv (a_\ell := *;\quad ?(-B \le a_\ell \le A))$$

$$f_{ctrl} \equiv \big(a_f := *;\quad ?(-B \le a_f \le -b)\big)$$

$$\cup \quad \big(?\mathbf{Safe}_\varepsilon;\quad a_f := *;\quad ?(-B \le a_f \le A)\big)$$

$$\cup \quad \big(?(v_f = 0);\; a_f := 0\big)$$

$$\mathbf{Safe}_\varepsilon \equiv x_f + \frac{v_f^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon v_f\right) < x_\ell + \frac{v_\ell^2}{2B}$$

$$dyn \equiv (t := 0;\; x_f' = v_f,\; v_f' = a_f,\; x_\ell' = v_\ell,\; v_\ell' = a_\ell, t' = 1$$

$$\&\; v_f \ge 0 \;\wedge\; v_\ell \ge 0 \;\wedge\; t \le \varepsilon)$$

Car Control: Global Lane Control Challenge

Challenge: Global lane dynamics

- All controllers for arbitrarily many differential equations respect separation globally on lane.

Challenge: Global lane dynamics

- All controllers for arbitrarily many differential equations respect separation globally on lane.
- Each car safe behind all others

Challenge: Global lane dynamics

- All controllers for arbitrarily many differential equations respect separation globally on lane.
- Each car safe behind all others

$$[(\forall i\, a(i) := ctrl;\ \forall i\, x(i)'' = a(i))^*]\, \forall i,j\ i \ll j$$

Car Control: Global Lane Control Challenge

$$\forall i : C\ i \ll \ell(i) \to [\mathtt{glc}](\forall i : C\ i \ll \ell^*(i))$$

Quantified Hybrid Program (Global lane control)

$$\begin{aligned}
\mathtt{glc} &\equiv (ctrl^n; dyn^n)^* \\
ctrl^n &\equiv \forall i : C\ (ctrl(i)) \\
ctrl(i) &\equiv \big(a(i) := *; ?(-B \le a(i) \le -b)\big) \\
&\quad \cup \quad \big(?\mathbf{Safe}_\varepsilon(i);\ a(i) := *;\ ?(-B \le a(i) \le A)\big) \\
&\quad \cup \quad \big(?(v(i) = 0);\ a(i) := 0\big) \\
\mathbf{Safe}_\varepsilon(i) &\equiv x(i) + \frac{v(i)^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon v(i)\right) < x(\ell(i)) + \frac{v(\ell(i))^2}{2B} \\
dyn^n &\equiv t := 0;\ \forall i : C\ (dyn(i), t' = 1\ \&\ v(i) \ge 0 \land t \le \varepsilon) \\
dyn(i) &\equiv x(i)' = v(i), v(i)' = a(i) \\
i \ll \ell^*(i) &\equiv [k := i;\ (k := \ell(k))^*] i \ll k
\end{aligned}$$

$$\forall i : C\ i \ll \ell(i) \to [\mathtt{glc}](\forall i : C\ i \ll \ell^*(i))$$

Quantified Hybrid Program (Global lane control)

$$\mathtt{glc} \equiv (ctrl^n; dyn^n)^*$$

$$ctrl^n \equiv \forall i : C\ (ctrl(i))$$

$$ctrl(i) \equiv (a(i) := *; ?(-B \le a(i) \le -b))$$
$$\cup\ (?\mathbf{Safe}_\varepsilon(i);\ a(i) := *;\ ?(-B \le a(i) \le A))$$
$$\cup\ (?(v(i) = 0);\ a(i) := 0)$$

$$\mathbf{Safe}_\varepsilon(i) \equiv x(i) + \frac{v(i)^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon v(i)\right) < x(\ell(i)) + \frac{v(\ell(i))^2}{2B}$$

$$dyn^n \equiv t := 0;\ \forall i : C\ (dyn(i), t' = 1\ \&\ v(i) \ge 0 \land t \le \varepsilon)$$

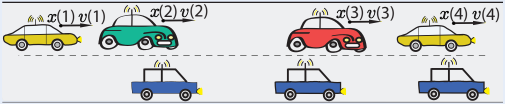
$$dyn(i) \equiv x(i)' = v(i), v(i)' = a(i)$$

$$i \ll \ell^*(i) \equiv [k := i;\ (k := \ell(k))^*] i \ll k$$

Car Control: Global Lane Control Challenge

$$\forall i : C \; i \ll \ell(i) \rightarrow [\mathtt{glc}](\forall i : C \; i \ll \ell^*(i))$$

Quantified Hybrid Program (Global lane control)



$$i \ll \ell^*(i) \equiv [k := i; \; (k := \ell(k))^*] i \ll k$$

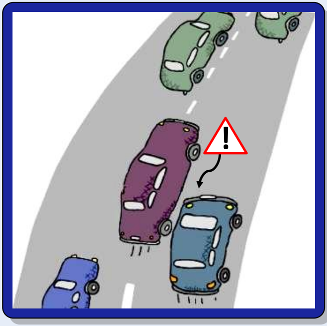
Car Control: Local Highway Control Challenge

Challenge: Local highway dynamics

- All controllers for arbitrarily many differential equations respect separation locally on highway.

Challenge: Local highway dynamics

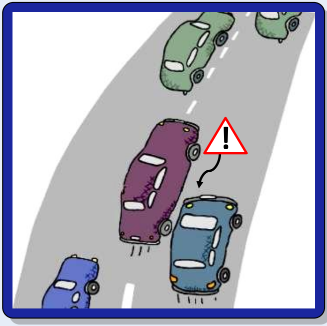
- All controllers for arbitrarily many differential equations respect separation locally on highway.
- For each lane: all controllers for the differential equations respect separation even if cars appear or disappear.



Car Control: Local Highway Control Challenge

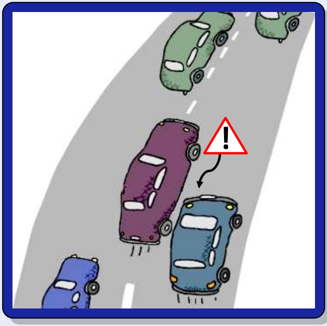
Challenge: Local highway dynamics

- All controllers for arbitrarily many differential equations respect separation locally on highway.
- For each lane: all controllers for the differential equations respect separation even if cars appear or disappear.
- Each car safe behind all others, even if new cars appear or disappear.



Challenge: Local highway dynamics

- All controllers for arbitrarily many differential equations respect separation locally on highway.
- For each lane: all controllers for the differential equations respect separation even if cars appear or disappear.
- Each car safe behind all others, even if new cars appear or disappear.



$$[(n := \texttt{new}\ C;\ \forall i\, a(i) := ctrl;\ \forall i\, x(i)'' = a(i))^*]\, \forall i,j\, i \ll j$$

Car Control: Local Highway Control Challenge

$$\forall i : C \; i \ll \ell(i) \rightarrow [\mathtt{glc}](\forall i : C \; i \ll \ell^*(i))$$

Quantified Hybrid Program (Local highway control)

$$\begin{aligned}
\mathtt{lhc} &\equiv (delete^*; create^*; ctrl^n; dyn^n)^* \\
create &\equiv n := new;\; ?(F(n) \ll n \land n \ll \ell(n)) \\
(n := new) &\equiv n := *;\; ?(\mathrm{E}(n) = 0);\; \mathrm{E}(n) := 1 \\
F(n) \ll n &\equiv \forall j : C \; (\ell(j) = n \rightarrow j \ll n) \\
delete &\equiv n := *;\; ?(\mathrm{E}(n) = 1);\; \mathrm{E}(n) := 0
\end{aligned}$$

Car Control: Local Highway Control Challenge

$$\forall i : C \; i \ll \ell(i) \rightarrow [\texttt{glc}](\forall i : C \; i \ll \ell^*(i))$$

Quantified Hybrid Program (Local highway control)

$$\begin{aligned}
\texttt{lhc} &\equiv (\mathit{delete}^*; \mathit{create}^*; \mathit{ctrl}^n; \mathit{dyn}^n)^* \\
\mathit{create} &\equiv n := \mathit{new};\; ?(F(n) \ll n \wedge n \ll \ell(n)) \\
(n := \mathit{new}) &\equiv n := *;\; ?(\mathrm{E}(n) = 0);\; \mathrm{E}(n) := 1 \\
F(n) \ll n &\equiv \forall j : C \; (\ell(j) = n \rightarrow j \ll n) \\
\mathit{delete} &\equiv n := *;\; ?(\mathrm{E}(n) = 1);\; \mathrm{E}(n) := 0
\end{aligned}$$

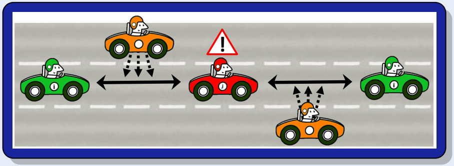
Car Control: Global Highway Control Challenge

Challenge: Global highway dynamics

- All controllers for arbitrarily many differential equations respect separation globally on highway.

Challenge: Global highway dynamics

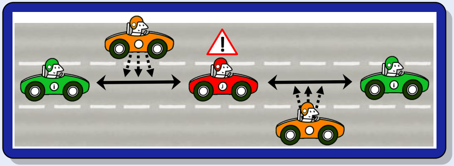
- All controllers for arbitrarily many differential equations respect separation globally on highway.
- All controllers for the differential equations respect separation even if cars switch lanes.



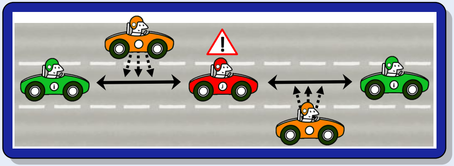
Car Control: Global Highway Control Challenge

Challenge: Global highway dynamics

- All controllers for arbitrarily many differential equations respect separation globally on highway.
- All controllers for the differential equations respect separation even if cars switch lanes.
- On all lanes, all car safe behind all others on their lanes, even if cars switch lanes.



Challenge: Global highway dynamics



- All controllers for arbitrarily many differential equations respect separation globally on highway.
- All controllers for the differential equations respect separation even if cars switch lanes.
- On all lanes, all car safe behind all others on their lanes, even if cars switch lanes.

$$[\forall l\,(n := \mathtt{new}\ C;\ \forall i\, a(i) := ctrl;\ \forall i\, x(i)'' = a(i))^*]\,\forall l\,\forall i,j\; i \ll j$$

$$\forall l : L \forall i : C_l\, i \ll \ell_l(i) \rightarrow$$
$$[(\forall l : L\, delete_l^*; \forall l : L\, new_l^*; \forall l : L\, ctrl_l^n; \forall l : L\, dyn_l^n)^*]\, \forall l : L \forall i : C_l\, i \ll \ell_l^*(i)$$

Quantified Hybrid Program (Global highway control)

$$\mathtt{ghc} \equiv (\forall l : L\, delete_l^*;\ \forall l : L\, new_l^*;\ \forall l : L\,,\, ctrl_l^n;\ \forall l : L\, dyn_l^n)^*$$

$$\forall l : L \forall i : C_l\, i \ll \ell_l(i) \rightarrow$$
$$[(\forall l : L\ delete_l^*; \forall l : L\ new_l^*; \forall l : L\ ctrl_l^n; \forall l : L\ dyn_l^n)^*]\ \forall l : L \forall i : C_l\, i \ll \ell_l^*(i)$$

Quantified Hybrid Program (Global highway control)

$$\mathrm{ghc} \equiv (\forall l : L\ delete_l^*;\ \forall l : L\ new_l^*;\ \forall l : L\,,ctrl_l^n;\ \forall l : L\ dyn_l^n)^*$$

Outline Background

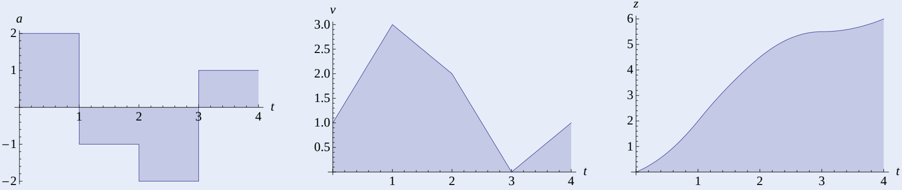
Q: I want to verify trains

Challenge

Q: I want to verify trains A: Hybrid systems

Challenge (Hybrid Systems)

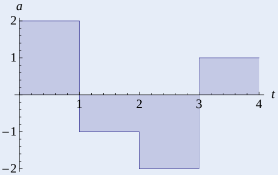
- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)

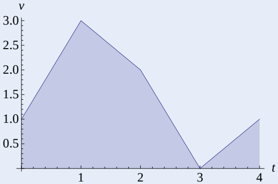


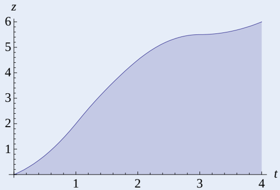
Q: I want to verify trains A: Hybrid systems Q: But there's uncertainties!

Challenge (Hybrid Systems)

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

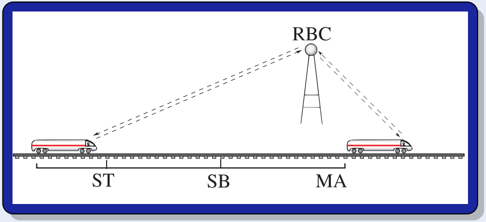






Q: I want to verify uncertain trains

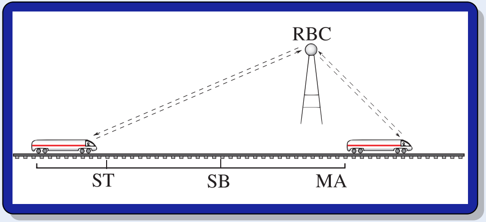
Challenge

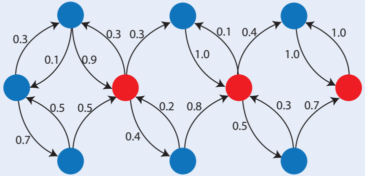


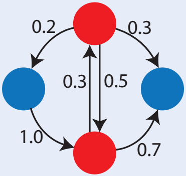
Q: I want to verify uncertain trains A: Markov chains

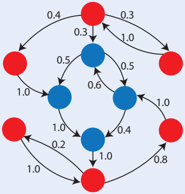
Challenge (Probabilistic Systems)

- Directed graph
  (Countable state space)
- Weighted edges
  (Transition probabilities)





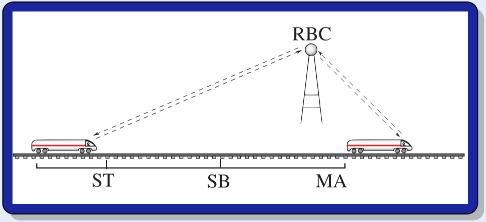


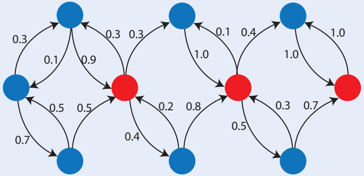


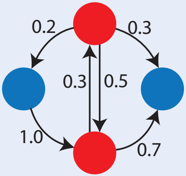
Q: I want to verify uncertain trains A: Markov chains Q: But trains move!

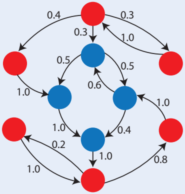
Challenge (Probabilistic Systems)

- Directed graph
  (Countable state space)
- Weighted edges
  (Transition probabilities)



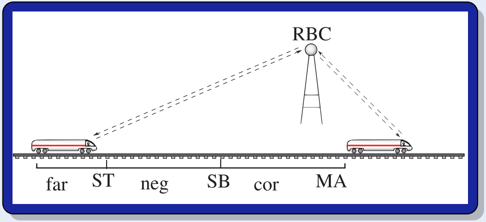






Q: I want to verify uncertain systems

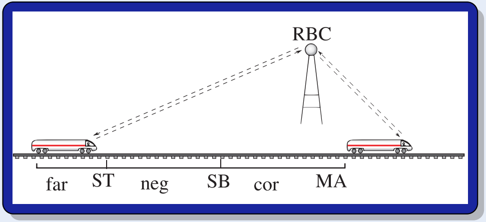
Challenge



Q: I want to verify uncertain systems A: Stochastic hybrid systems

Challenge (Stochastic Hybrid Systems)

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

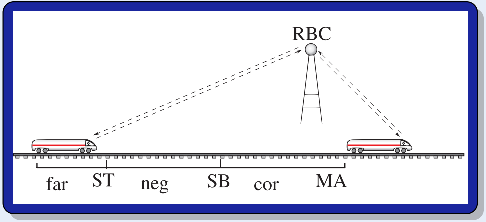


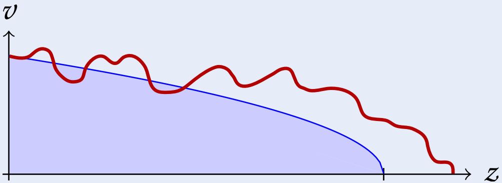
Q: I want to verify uncertain systems A: Stochastic hybrid systems

Challenge (Stochastic Hybrid Systems)

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

- Discrete stochastic (lossy communication)
- Continuous stochastic (wind, track)



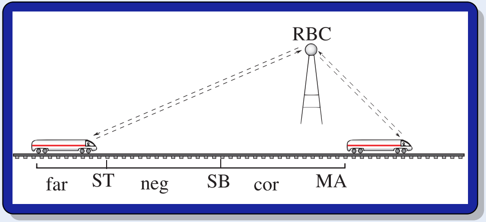


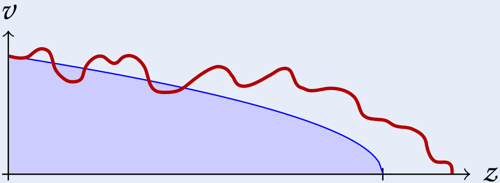
Q: I want to verify uncertain systems A: Stochastic hybrid systems Q: How?

Challenge (Stochastic Hybrid Systems)

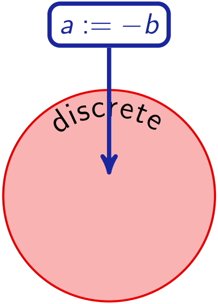
- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)
- Stochastic dynamics (uncertainty)

- Discrete stochastic (lossy communication)
- Continuous stochastic (wind, track)

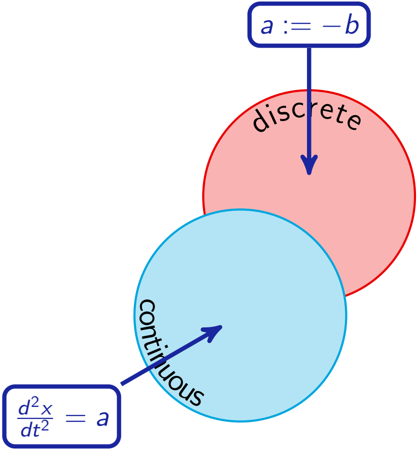




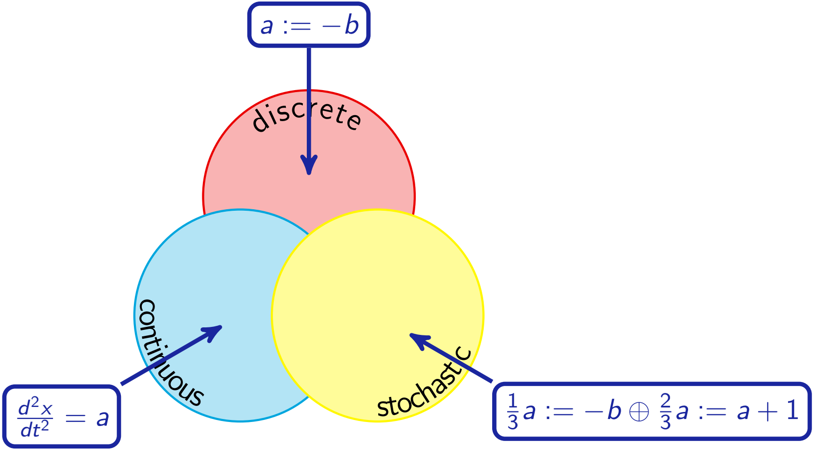
Model for Stochastic Hybrid Systems

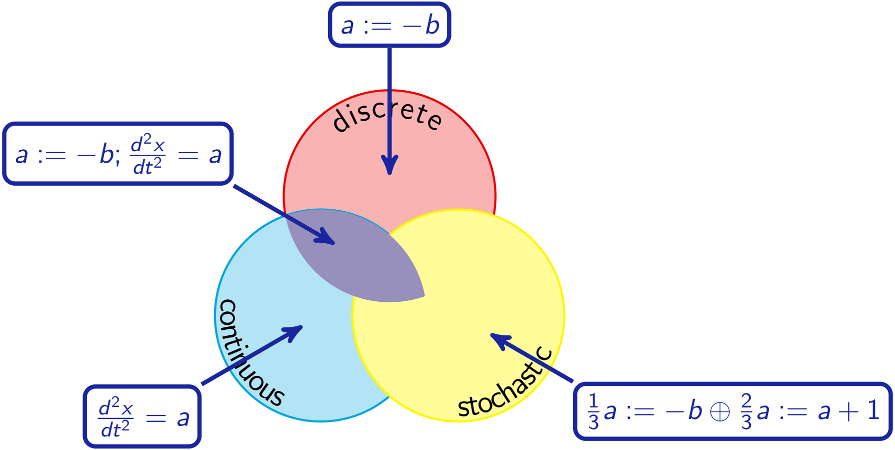


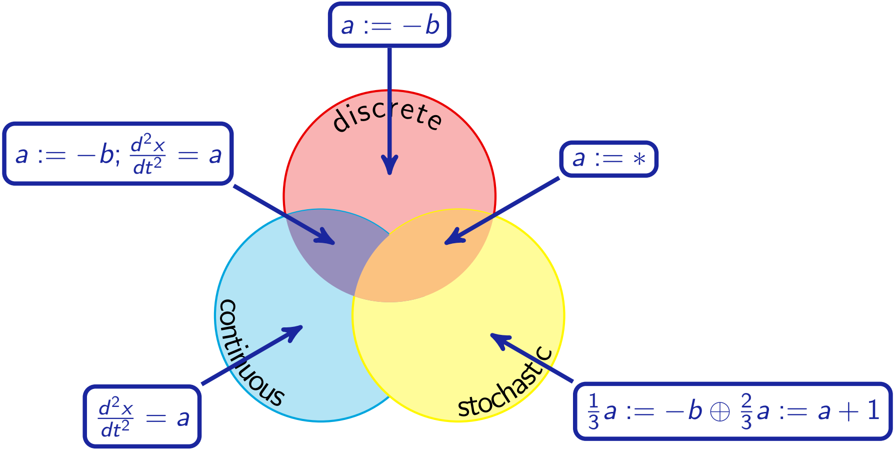
Model for Stochastic Hybrid Systems



Model for Stochastic Hybrid Systems

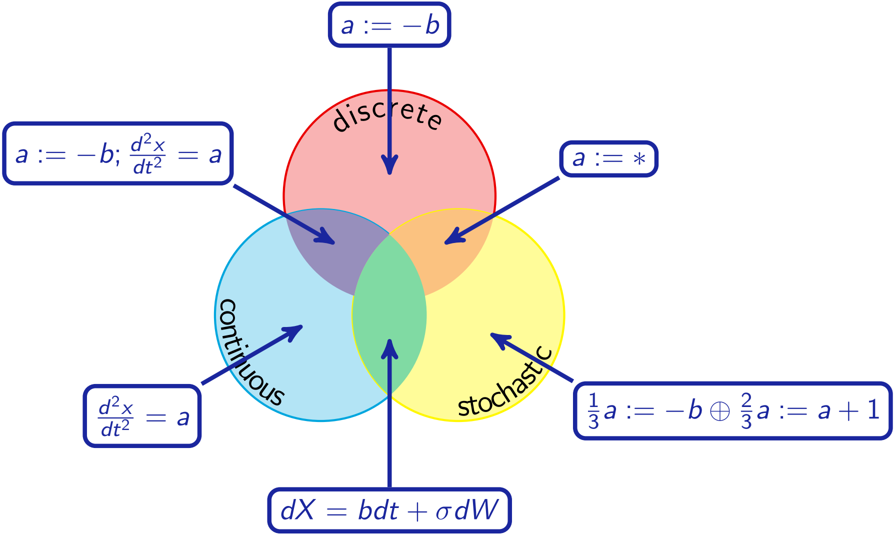



$a := -b$
$a := -b; \frac{d^2x}{dt^2} = a$
discrete
continuous
stochastic
$\frac{d^2x}{dt^2} = a$
$\frac{1}{3}a := -b \oplus \frac{2}{3}a := a + 1$

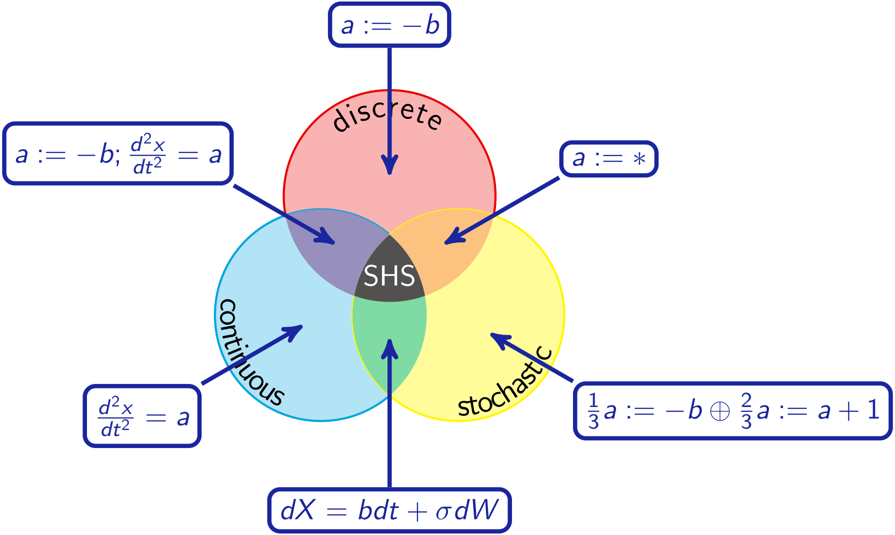


$a := -b$
$a := -b; \frac{d^2x}{dt^2} = a$
$a := *$
discrete
continuous
stochastic
$\frac{d^2x}{dt^2} = a$
$\frac{1}{3}a := -b \oplus \frac{2}{3}a := a + 1$

Model for Stochastic Hybrid Systems

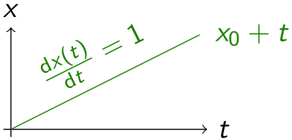


Model for Stochastic Hybrid Systems



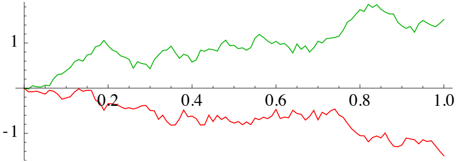
Definition (Ordinary differential equation (ODE))

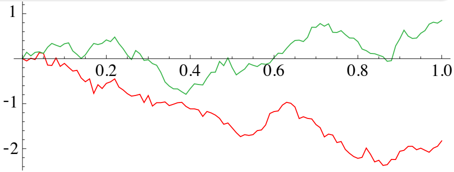
$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = b(x(t)) \quad x(0) = x_0$$



Definition (Itō stochastic differential equation (SDE))

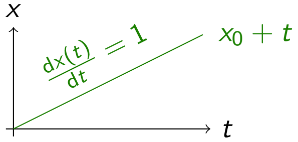
$$dX_t = b(X_t)dt + \sigma(X_t)dW_t \quad X_0 = Z$$





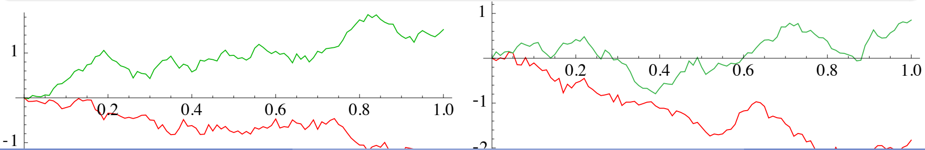
Definition (Ordinary differential equation (ODE))

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = b(x(t)) \quad x(0) = x_0$$



Definition (Itō stochastic differential equation (SDE))

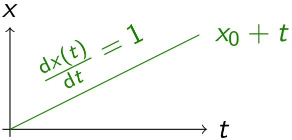
$$X_s = Z + \int_0^s dX_t = Z + \int_0^s b(X_t)dt + \int_0^s \sigma(X_t)dW_t$$



Stochastic Differential Equations (SDE)

Definition (Ordinary differential equation (ODE))

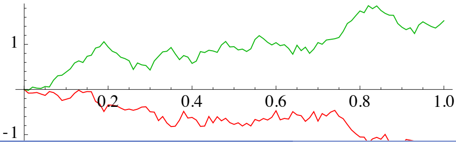
$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = b(x(t)) \quad x(0) = x_0$$

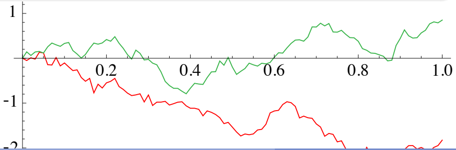


Calculus

Definition (Itō stochastic differential equation (SDE))

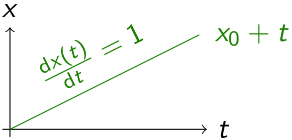
$$X_s = Z + \int_0^s dX_t = Z + \int_0^s b(X_t)dt + \int_0^s \sigma(X_t)dW_t$$

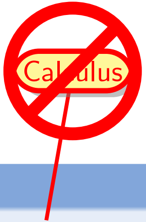




Definition (Ordinary differential equation (ODE))

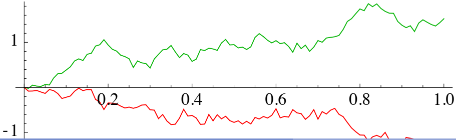
$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = b(x(t)) \quad x(0) = x_0$$

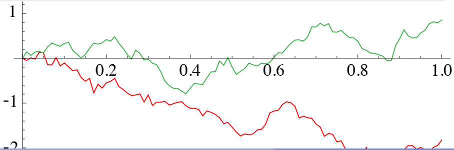




Definition (Itō stochastic differential equation (SDE))

$$X_s = Z + \int_0^s dX_t = Z + \int_0^s b(X_t)dt + \int_0^s \sigma(X_t)dW_t$$





Brownian Motion is Extremely Complex

Definition (Brownian motion W $\Rightarrow$ end of calculus)

1. $W_0 = 0$ (start at 0)
2. W_t almost surely continuous
3. $W_t - W_s \sim \mathcal{N}(0, t-s)$ (independent normal increments)

$\Rightarrow$ a.s. continuous everywhere but nowhere differentiable

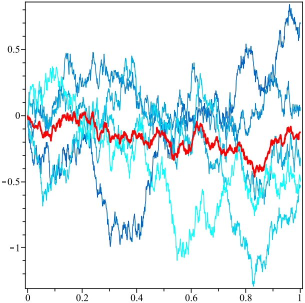
$\Rightarrow$ a.s. unbounded variation, $\notin$ FV, nonmonotonic on every interval

Definition (Brownian motion W $\Rightarrow$ end of calculus)

1. $W_0 = 0$ (start at 0)
2. W_t almost surely continuous
3. $W_t - W_s \sim \mathcal{N}(0, t-s)$ (independent normal increments)

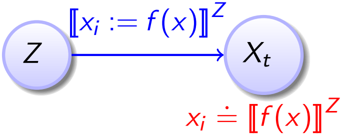
$\Rightarrow$ a.s. continuous everywhere but nowhere differentiable

$\Rightarrow$ a.s. unbounded variation, $\notin$ FV, nonmonotonic on every interval



Definition (Stochastic hybrid program α)

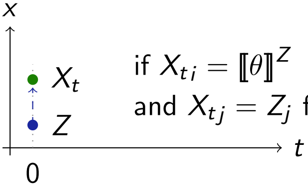
$x := \theta$	(assignment)	jump & test
$x := *$	(random assignment)	
$?H$	(conditional execution)	
$dx = bdt + \sigma dW \,\&\, H$	(SDE)	
$\alpha; \beta$	(seq. composition)	algebra
$\lambda\alpha \oplus \nu\beta$	(convex combination)	
α^*	(nondet. repetition)	



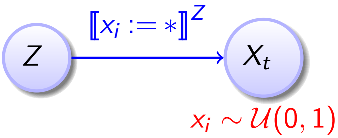
Definition (Stochastic hybrid program α: process semantics)

$$[\![x_i := \theta]\!]^Z = \hat{Y} \quad Y(\omega)_i = [\![\theta]\!]^{Z(\omega)} \text{ and } Y_j = Z_j \text{ (for } j \neq i)$$

$$(\!|x_i := \theta|\!)^Z = 0$$



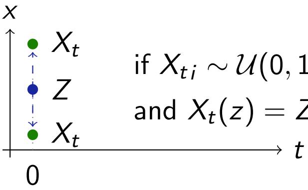
if $X_{t\,i} = [\![\theta]\!]^Z$
and $X_{t\,j} = Z_j$ for $j \neq i$



Definition (Stochastic hybrid program α: process semantics)

$$[\![x_i := *]\!]^Z = \hat{U} \qquad U_i \sim \mathcal{U}(0,1) \text{ i.i.d. } \mathcal{F}_0\text{-measurable}$$

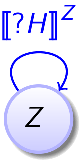
$$(\!|x_i := *|\!)^Z = 0$$



if $X_{t\,i} \sim \mathcal{U}(0,1)$

and $X_t(z) = Z(z)$ for $z \neq x$

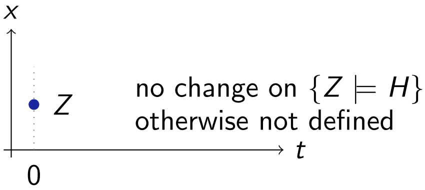
Stochastic Hybrid Program: Process Semantics

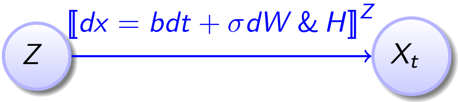


on $\{Z \models H\}$

Definition (Stochastic hybrid program α: process semantics)

$$[\![?H]\!]^Z = \hat{Z} \quad \text{on the event } \{Z \models H\}$$
$$(\!|?H|\!)^Z = 0$$

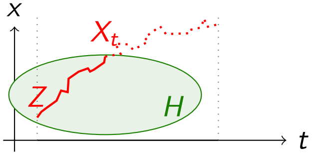




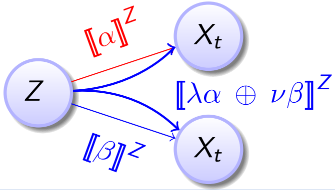
Definition (Stochastic hybrid program α: process semantics)

$$[\![dx = bdt + \sigma dW \;\&\; H]\!]^Z \text{ solves } dX = [\![b]\!]^X dt + [\![\sigma]\!]^X dB_t, X_0 = Z$$

$$(\!|dx = bdt + \sigma dW \;\&\; H|\!)^Z = \inf\{t \geq 0 \;:\; X_t \notin H\}$$



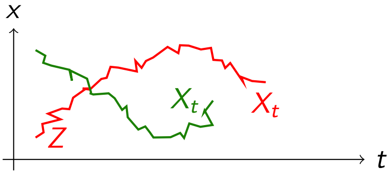
$dx = bdt + \sigma dW \;\&\; H$

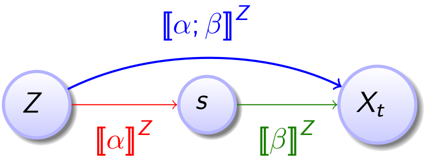


Definition (Stochastic hybrid program α: process semantics)

$$[\![\lambda\alpha \oplus \nu\beta]\!]^Z = \mathcal{I}_{U\le\lambda}[\![\alpha]\!]^Z + \mathcal{I}_{U>\lambda}[\![\beta]\!]^Z = \begin{cases} [\![\alpha]\!]^Z & \text{on event } \{U \le \lambda\} \\ [\![\beta]\!]^Z & \text{on event } \{U > \lambda\} \end{cases}$$

$(\!|\lambda\alpha \oplus \nu\beta|\!)^Z = \mathcal{I}_{U\le\lambda}(\!|\alpha|\!)^Z + \mathcal{I}_{U>\lambda}(\!|\beta|\!)^Z$ with i.i.d. $U \sim \mathcal{U}(0,1)$, $\mathcal{F}_0$-meas

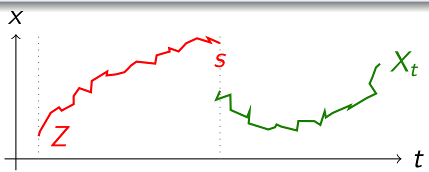


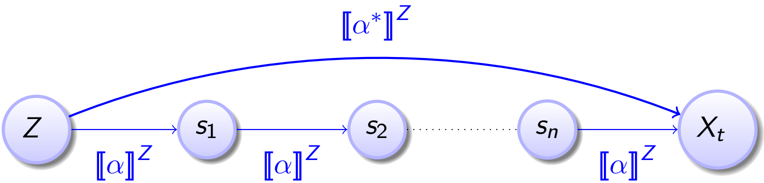


Definition (Stochastic hybrid program α: process semantics)

$$\llbracket \alpha;\beta \rrbracket_t^Z = \begin{cases} \llbracket \alpha \rrbracket_t^Z & \text{on event } \{t < \llparenthesis \alpha \rrparenthesis^Z\} \\ \llbracket \beta \rrbracket_{t-\llparenthesis \alpha \rrparenthesis^Z}^{\llbracket \alpha \rrbracket^Z_{\llparenthesis \alpha \rrparenthesis^Z}} & \text{on event } \{t \geq \llparenthesis \alpha \rrparenthesis^Z\} \end{cases}$$

$$\llparenthesis \alpha;\beta \rrparenthesis^Z = \llparenthesis \alpha \rrparenthesis^Z + \llparenthesis \beta \rrparenthesis^{\llbracket \alpha \rrbracket^Z_{\llparenthesis \alpha \rrparenthesis^Z}}$$

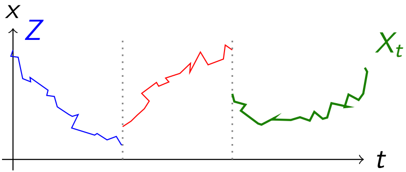


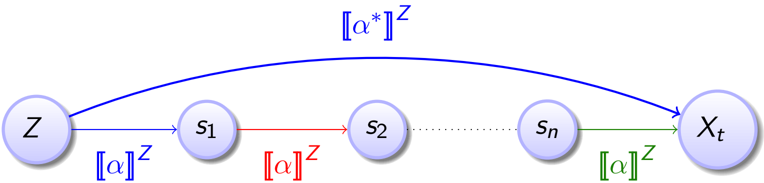


Definition (Stochastic hybrid program α: process semantics)

$$[\![\alpha^*]\!]^Z_t = [\![\alpha^n]\!]^Z_t \text{ on event } \{(\!|\alpha^n|\!)^Z > t\}$$

$$(\!|\alpha^*|\!)^Z = \lim_{n\to\infty} (\!|\alpha^n|\!)^Z$$

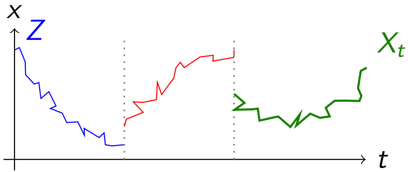




Definition (Stochastic hybrid program α: process semantics)

$$[\![\alpha^*]\!]^Z_t = [\![\alpha^n]\!]^Z_t \text{ on event } \{(\!|\alpha^n|\!)^Z > t\}$$

$$(\!|\alpha^*|\!)^Z = \lim_{n\to\infty} (\!|\alpha^n|\!)^Z \qquad \text{monotone!}$$



Stochastic Differential Dynamic Logic Sd$\mathcal{L}$: Syntax

Definition (Sd$\mathcal{L}$ term f)

F	(primitive measurable function, e.g., characteristic $\mathcal{I}_A$)
$\lambda f + \nu g$	(linear term)
Bf	(scalar term for boolean term B)
$\langle\alpha\rangle f$	(reachable)

Definition (Sd$\mathcal{L}$ formula ϕ)

$$\phi \ ::= \ f \leq g \mid f = g$$

Definition (Measurable semantics)

Definition (Measurable semantics)

$$[\![F]\!]^Z = F^\ell(Z) \text{ i.e., } [\![F]\!]^Z(\omega) = F^\ell(Z(\omega))$$

Stochastic Differential Dynamic Logic Sd$\mathcal{L}$: Semantics

Definition (Measurable semantics)

$$[\![F]\!]^Z = F^\ell(Z) \text{ i.e., } [\![F]\!]^Z(\omega) = F^\ell(Z(\omega))$$

$$[\![\lambda f + \nu g]\!]^Z = \lambda[\![f]\!]^Z + \nu[\![g]\!]^Z$$

Definition (Measurable semantics)

$$[\![F]\!]^Z = F^\ell(Z) \text{ i.e., } [\![F]\!]^Z(\omega) = F^\ell(Z(\omega))$$

$$[\![\lambda f + \nu g]\!]^Z = \lambda[\![f]\!]^Z + \nu[\![g]\!]^Z$$

$$[\![Bf]\!]^Z = [\![B]\!]^Z * [\![f]\!]^Z \text{ i.e., } [\![Bf]\!]^Z(\omega) = [\![B]\!]^Z(\omega)[\![f]\!]^Z(\omega)$$

Definition (Measurable semantics)

$$[\![F]\!]^Z = F^\ell(Z) \text{ i.e., } [\![F]\!]^Z(\omega) = F^\ell(Z(\omega))$$

$$[\![\lambda f + \nu g]\!]^Z = \lambda[\![f]\!]^Z + \nu[\![g]\!]^Z$$

$$[\![Bf]\!]^Z = [\![B]\!]^Z * [\![f]\!]^Z \text{ i.e., } [\![Bf]\!]^Z(\omega) = [\![B]\!]^Z(\omega)[\![f]\!]^Z(\omega)$$

$$[\![\langle\alpha\rangle f]\!]^Z = \sup\{[\![f]\!]^{[\![\alpha]\!]_t^Z} : 0 \le t \le (\!|\alpha|\!)^Z\}$$

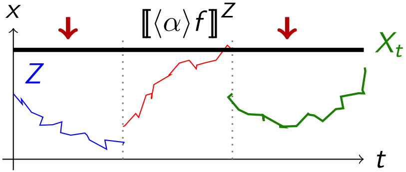
Definition (Measurable semantics)

$$[\![F]\!]^Z = F^\ell(Z) \text{ i.e., } [\![F]\!]^Z(\omega) = F^\ell(Z(\omega))$$

$$[\![\lambda f + \nu g]\!]^Z = \lambda[\![f]\!]^Z + \nu[\![g]\!]^Z$$

$$[\![Bf]\!]^Z = [\![B]\!]^Z * [\![f]\!]^Z \text{ i.e., } [\![Bf]\!]^Z(\omega) = [\![B]\!]^Z(\omega)[\![f]\!]^Z(\omega)$$

$$[\![\langle\alpha\rangle f]\!]^Z = \sup\{[\![f]\!]^{[\![\alpha]\!]_t^Z} \ : \ 0 \le t \le (\!|\alpha|\!)^Z\}$$



Well-definedness of Sd$\mathcal{L}$ Semantics

Theorem (Measurable)

$[\![f]\!]^Z$ *is a random variable (i.e., measurable) for any random variable* Z *and* Sd$\mathcal{L}$ *term* f.

Well-definedness of Sd$\mathcal{L}$ Semantics

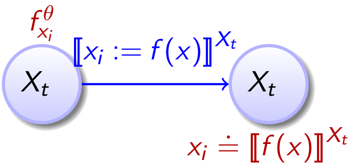
Theorem (Measurable)

$[\![f]\!]^Z$ *is a random variable (i.e., measurable) for any random variable Z and* Sd$\mathcal{L}$ *term f.*

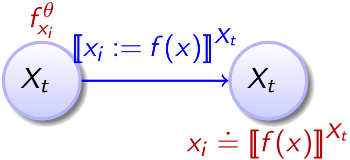
Corollary (Pushforward measure well-defined for Borel-measurable S)

$$S \mapsto P(([\![f]\!]^Z)^{-1}(S)) = P(\{\omega \in \Omega \;:\; [\![f]\!]^Z(\omega) \in S\}) = P([\![f]\!]^Z \in S)$$

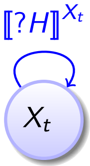
$$\langle x_i := \theta \rangle f \;=\; f_{x_i}^{\theta}$$



$$\langle x_i := \theta \rangle f \; = \; f_{x_i}^{\theta}$$



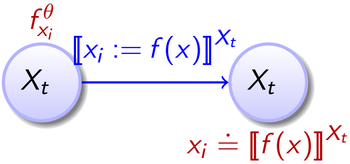
$$\langle ?H \rangle f \; = \; Hf$$



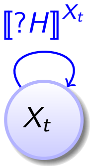
on $\{X_t \models H\}$

Proof Calculus: Stochastic Differential Dynamic Logic

$$\langle x_i := \theta \rangle f = f_{x_i}^{\theta}$$



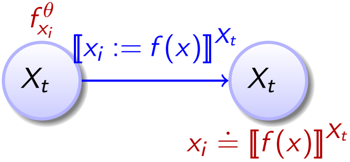
$$\langle ?H \rangle f = Hf$$



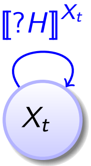
on $\{X_t \models H\}$

$$\langle \alpha \rangle (\lambda f) = \lambda \langle \alpha \rangle f$$

$$\langle x_i := \theta \rangle f \;=\; f_{x_i}^{\theta}$$



$$\langle ?H \rangle f \;=\; Hf$$



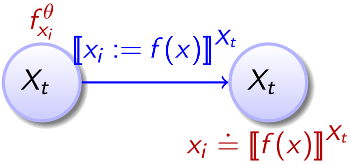
on $\{X_t \models H\}$

$$\langle \alpha \rangle (\lambda f) \;=\; \lambda \langle \alpha \rangle f$$

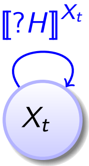
$$\langle \alpha \rangle (\lambda f + \nu g) \;\leq\; \lambda \langle \alpha \rangle f + \nu \langle \alpha \rangle g$$

Proof Calculus: Stochastic Differential Dynamic Logic

$$\langle x_i := \theta \rangle f \;=\; f_{x_i}^{\theta}$$



$$\langle ?H \rangle f \;=\; Hf$$



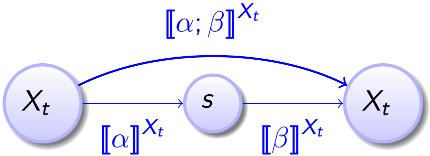
on $\{X_t \models H\}$

$$\langle \alpha \rangle (\lambda f) \;=\; \lambda \langle \alpha \rangle f$$

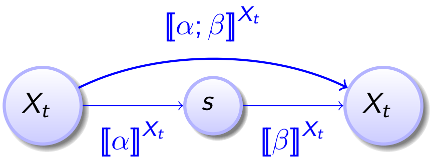
$$\langle \alpha \rangle (\lambda f + \nu g) \;\leq\; \lambda \langle \alpha \rangle f + \nu \langle \alpha \rangle g$$

$$f \leq g \models \langle \alpha \rangle f \leq \langle \alpha \rangle g$$

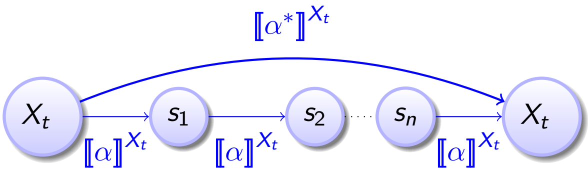
$$\langle \alpha;\beta\rangle f \;\le\; \langle \alpha\rangle (f \sqcup \langle \beta\rangle f)$$



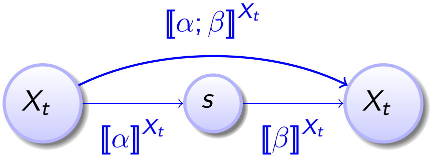
$\langle\alpha;\beta\rangle f \leq \langle\alpha\rangle(f \sqcup \langle\beta\rangle f)$



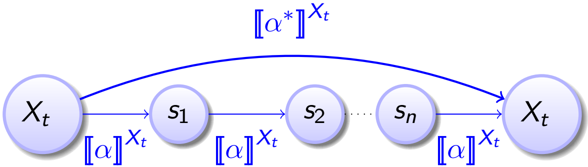
$\langle\alpha\rangle f \leq f \vDash \langle\alpha^*\rangle f \leq f$



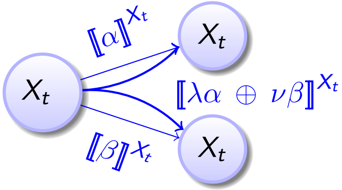
$\langle\alpha;\beta\rangle f \;\le\; \langle\alpha\rangle(f \sqcup \langle\beta\rangle f)$



$\langle\alpha\rangle f \le f \vDash \langle\alpha^*\rangle f \le f$



$$P(\langle\lambda\alpha \oplus \nu\beta\rangle f \in S)$$
$$= \lambda P(\langle\alpha\rangle f \in S)$$
$$+ \nu P(\langle\beta\rangle f \in S)$$



Soundness

Theorem (Soundness)

1. *Rules are globally sound pathwise, i.e.,* $f_i \leq g_i \vDash f \leq g$ *holds for each initial* Z *pathwise for each* $\omega \in \Omega$
2. $\langle \oplus \rangle$ *is sound in distribution*

Soundness

Theorem (Soundness)

1. *Rules are globally sound pathwise, i.e.,* $f_i \leq g_i \vDash f \leq g$ *holds for each initial* Z *pathwise for each* $\omega \in \Omega$
2. $\langle\oplus\rangle$ *is sound in distribution*

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle \phi \geq \lambda) \leq p} \quad \textit{sound}$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle\phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x \tau < \infty}{\Longrightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s)ds$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle\phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x\tau<\infty}{\Rightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s)ds$$

Theorem (Differential generator for SDE solution and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$A\phi = L\phi := b\nabla f + \frac{\sigma\sigma^T}{2}\nabla\nabla f$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C^2_C(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle \phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C^2_C(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x\tau<\infty}{\Rightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s)ds$$

Theorem (Differential generator for SDE solution and $\phi \in C^2_C(\mathbb{R}^d, \mathbb{R})$)

$$A\phi = L\phi := b\nabla f + \frac{\sigma\sigma^T}{2}\nabla\nabla f = \sum_i b_i \frac{\partial f}{\partial x_i} + \frac{1}{2}\sum_{i,j}(\sigma\sigma^T)_{i,j}\frac{\partial^2 f}{\partial x_i \partial x_j}$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ compact support on H (e.g., H bounded)

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle\phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x\tau < \infty}{\Rightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s) ds$$

$$A\phi(X_s) = L\phi(X_s) \leq 0 \text{ on } H \quad \Rightarrow \quad E^x \phi(X_\tau) \leq \phi(x) \forall x, \tau$$

$$\Rightarrow \quad P^x\text{-a.s. } E^x(\phi(X_t) | \mathcal{F}_s) = E^{X_s} \phi(X_{t-s}) \leq \phi(X_s)$$

$$\Rightarrow \quad X_t \text{ supermartingale}$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle \phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x \tau < \infty}{\Rightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s) ds$$

Theorem (Doob maximal martingale ineq., càdlàg supermartingale)

$$\forall\ f \geq 0, \lambda > 0 \quad P\left(\sup_{t \geq 0} f(X_t) \geq \lambda \,|\, \mathcal{F}_0\right) \leq \frac{Ef(X_0)}{\lambda}$$

Soundness

Theorem (Stochastic Differential Invariants)

Let $\lambda > 0$, $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$ *compact support on* H *(e.g.,* H *bounded)*

$$\frac{\langle\alpha\rangle(H \to \phi) \leq \lambda p \quad H \to \phi \geq 0 \quad H \to Lf \leq 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle\phi \geq \lambda) \leq p} \quad \textit{sound}$$

Theorem (Dynkin for càdlàg strong Markov X_t and $\phi \in C_C^2(\mathbb{R}^d, \mathbb{R})$)

$$Af(x) := \lim_{t \searrow 0} \frac{E^x f(X_t) - f(x)}{t} \overset{E^x\tau<\infty}{\Rightarrow} E^x f(X_\tau) = f(x) + E^x \int_0^\tau Af(X_s)ds$$

Theorem (Doob maximal martingale ineq., càdlàg supermartingale)

$$\forall\ f \geq 0, \lambda > 0 \quad P\left(\sup_{t \geq 0} f(X_t) \geq \lambda \,|\, \mathcal{F}_0\right) \leq \frac{Ef(X_0)}{\lambda} \leq \frac{\lambda p}{\lambda} = p$$

Proof Example

$$\frac{\langle\alpha\rangle(H \to \phi) \le \lambda p \quad H \to \phi \ge 0 \quad H \to Lf \le 0}{P(\langle\alpha\rangle\langle dx = bdt + \sigma dW \,\&\, H\rangle\phi \ge \lambda) \le p}$$

$$\langle ?x^2 + y^2 \le \frac{1}{3}\rangle(H \to \phi) = \left(H \to x^2 + y^2 \le \frac{1}{3}\right)(x^2 + y^2) \le 1 * \frac{1}{3}$$

$\phi \equiv x^2 + y^2 \ge 0$ with $H \equiv x^2 + y^2 < 10$

$$\frac{L\phi = \frac{1}{2}\left(-x\frac{\partial\phi}{\partial x} - y\frac{\partial\phi}{\partial y} + y^2\frac{\partial^2\phi}{\partial x^2} - 2xy\frac{\partial^2\phi}{\partial x\partial y} + x^2\frac{\partial^2\phi}{\partial y^2}\right) \le 0}{P(\langle ?x^2 + y^2 \le \frac{1}{3}; dx = -\frac{x}{2}dt - ydW, dy = -\frac{y}{2}dt + xdW \,\&\, H\rangle x^2 + y^2 \ge 1)}$$

$\le$ (by **??**)

$$P(\langle ?x^2 + y^2 \le \frac{1}{3}\rangle\langle dx = -\frac{x}{2}dt - ydW, dy = -\frac{y}{2}dt + xdW \,\&\, H\rangle x^2 + y^2 \ge 1$$

$$\le \frac{1}{3}$$