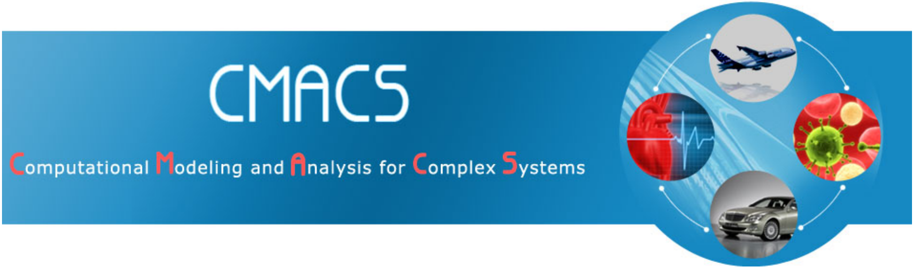
Verification of Nonlinear Models and Compositional Models

André Platzer

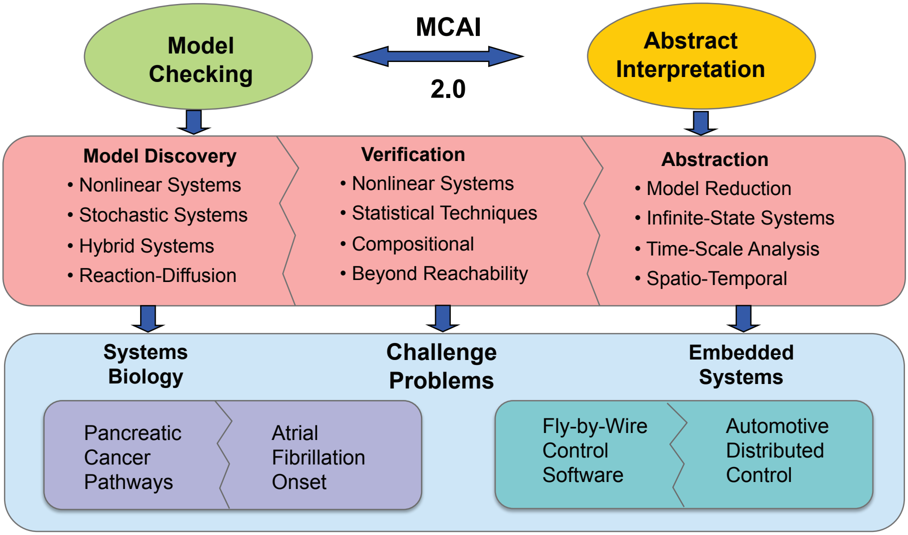
Carnegie Mellon University, Computer Science Department, Pittsburgh, PA

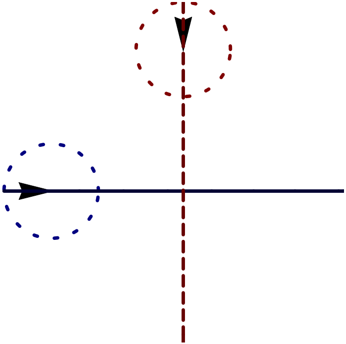


Outline

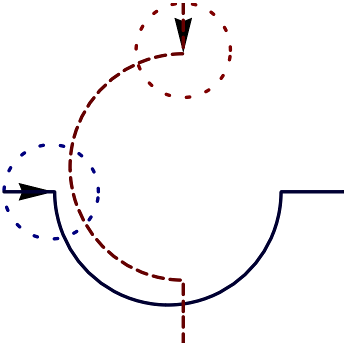
1. CMACS Context
2. Approximation in Model Checking
   - Bounded Flow Approximation
   - Continuous Image Computation
   - Probabilistic Model Checking
   - Differential Invariants
3. Compositional Verification of Hybrid Systems
   - Compositionality in Verification
   - Discrete Induction
   - Differential Induction
4. Computing Differential Invariants by Combining Local Fixedpoints
   - Local Fixedpoints Iteration
   - Global Fixedpoints & Fixedpoint Loop Combinations
5. Collision Avoidance Maneuvers in Air Traffic Control
5. Summary & Plans

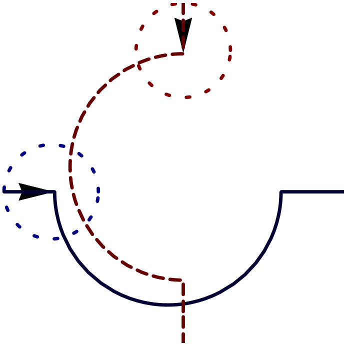
Verification: Nonlinear Models & Compositional Models





Verification of Nonlinear Hybrid Systems

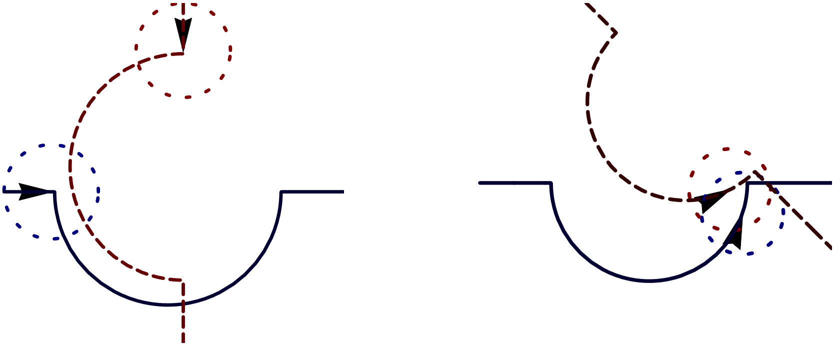




Hybrid Systems

continuous evolution along differential equations + discrete change

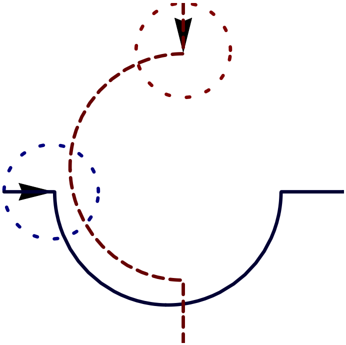
Verification of Nonlinear Hybrid Systems

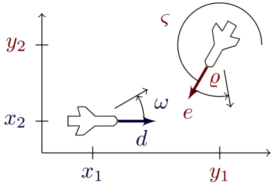


Hybrid Systems

continuous evolution along differential equations + discrete change

Verification of Nonlinear Hybrid Systems



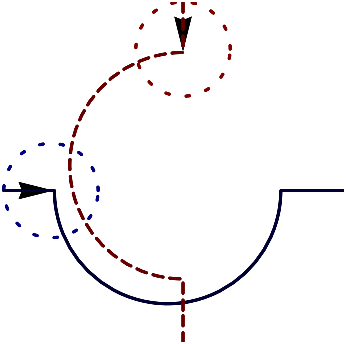


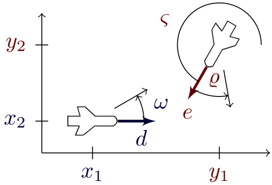
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varrho - \omega \end{bmatrix}$$

Hybrid Systems

continuous evolution along differential equations + discrete change

Verification of Nonlinear Hybrid Systems



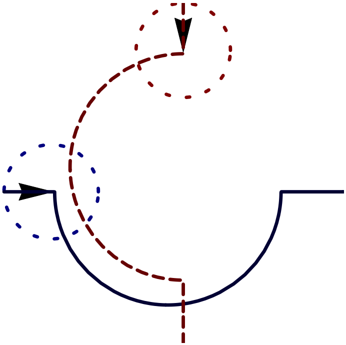


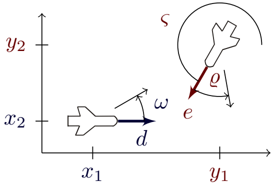
$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varrho - \omega \end{bmatrix}$$

Example ("Solving" differential equations)

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

Verification of Nonlinear Hybrid Systems





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

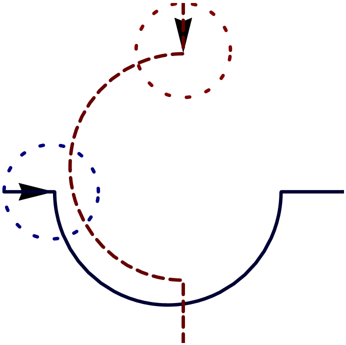
Example ("Solving" differential equations)

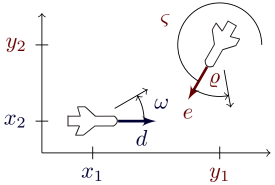
$$\forall t{\geq}0 \quad \frac{1}{\omega\varrho}\big(x_1\omega\varrho\cos t\omega - v_2\omega\cos t\omega\sin\vartheta + v_2\omega\cos t\omega\cos t\varrho\sin\vartheta - v_1\varrho\sin t\omega$$

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

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

Verification of Nonlinear Hybrid Systems





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

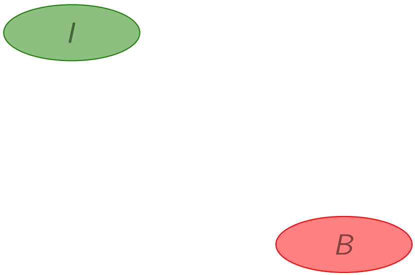
Symbolic Verification	Numerical Verification
× constant/nilpotent systems	✓ nonlinear systems
× otherwise "no" solutions	× approximation errors
✓ sound	× sound ...?

Outline

1. CMACS Context
2. Approximation in Model Checking
   - Bounded Flow Approximation
   - Continuous Image Computation
   - Probabilistic Model Checking
   - Differential Invariants
3. Compositional Verification of Hybrid Systems
   - Compositionality in Verification
   - Discrete Induction
   - Differential Induction
4. Computing Differential Invariants by Combining Local Fixedpoints
   - Local Fixedpoints Iteration
   - Global Fixedpoints & Fixedpoint Loop Combinations
5. Collision Avoidance Maneuvers in Air Traffic Control
5. Summary & Plans

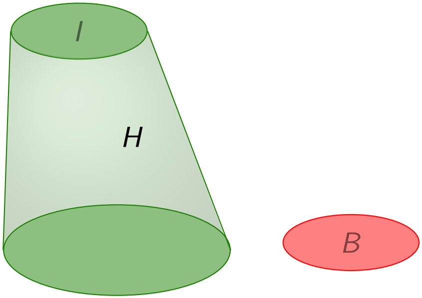
AMC(B reachable from I in H):

1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC(B reachable from I in H):

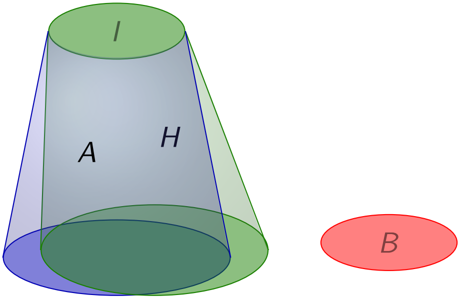
1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC: Approximation Refinement Model Checking

AMC(B reachable from I in H):

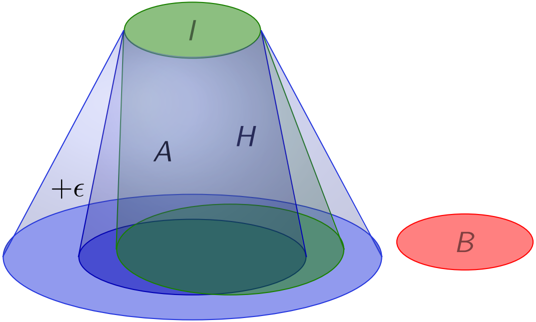
1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC: Approximation Refinement Model Checking

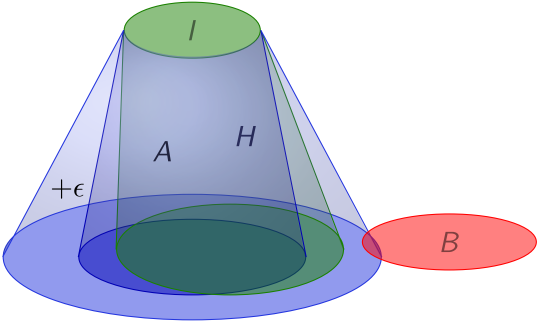
AMC(B reachable from I in H):

1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC(B reachable from I in H):

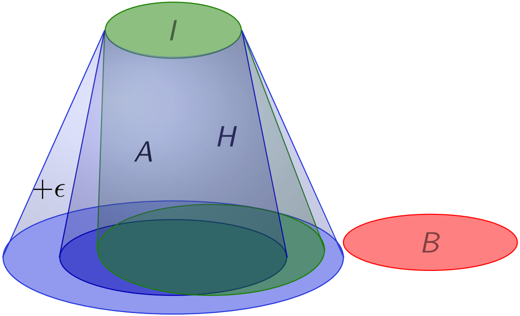
1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC: Approximation Refinement Model Checking

AMC(B reachable from I in H):

1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe



AMC: Exact Image Computation

AMC(B reachable from I in H):

1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe

Proposition

check and blur can be implemented for

- *I and B semialgebraic*
- *A with polynomial flows over $\mathbb{R}$*
- *+Piecewise definitions*
- *+Rational extensions (e.g. multivariate rational splines)*

AMC: Image Approximation

AMC(B reachable from I in H):

1. $A :=$ approx(H) uniformly
2. blur by uniform approximation error $+\epsilon$
3. check(B reachable from I in $A + \epsilon$)
4. B not reachable $\Rightarrow$ H safe

Proposition

approx exists for all uniform errors $\epsilon > 0$ when

- *using polynomials to build A*
- *Flows $\varphi \in C(D, \mathbb{R}^n)$ of H*
- *$D \subset \mathbb{R} \times \mathbb{R}^n$ compact closure of an open set*

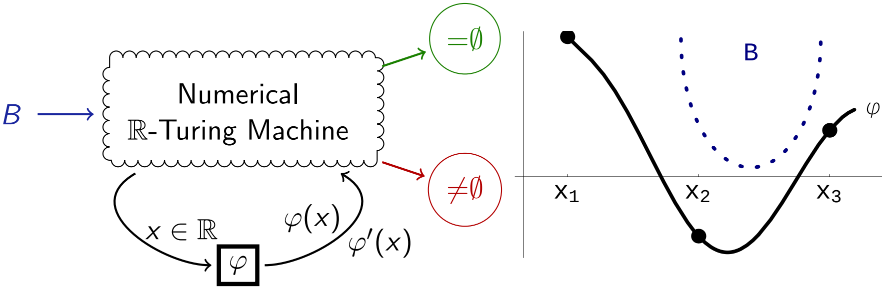
$\mathcal{A}$ Bounded Flow Approximation

Proposition (Effective Weierstraß approximation)

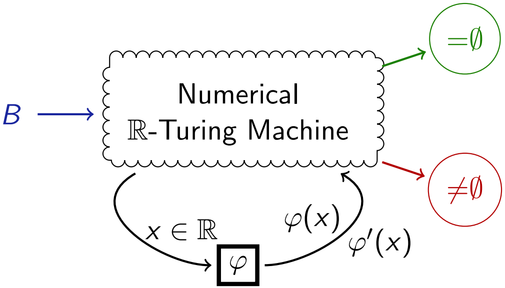
- *Flows* $\varphi \in C^1(D, \mathbb{R}^n)$
- *Bounds* $b := \max_{x \in D} \|\varphi'(x)\|$

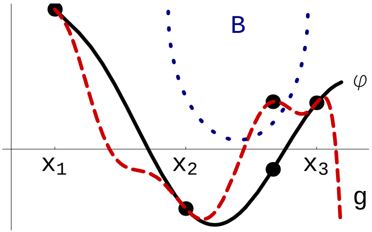
$\Rightarrow$ *approx computable, hence image computation decidable*

Continuous Image Computation

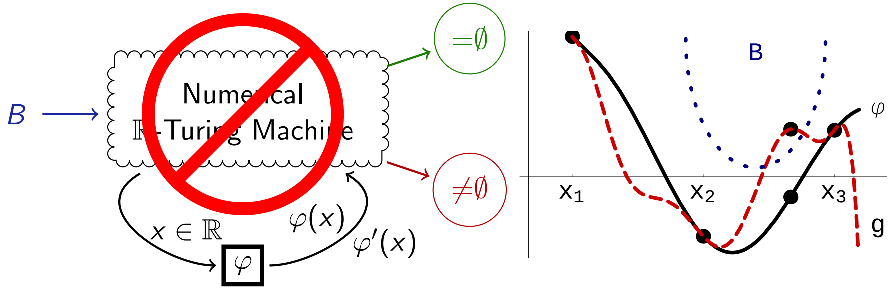


Continuous Image Computation



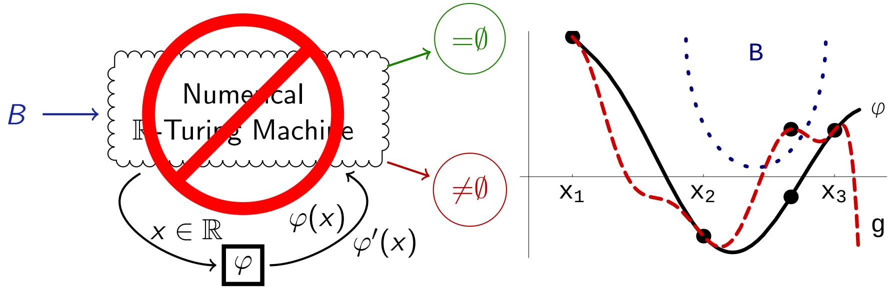


Continuous Image Computation



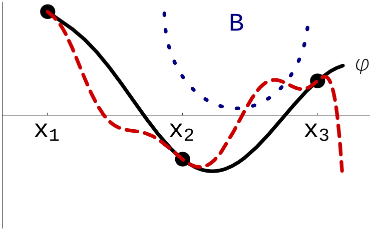
Proposition (Image computation undecidable for...)

- *arbitrarily effective flow* $\varphi \in C^k(D \subseteq \mathbb{R}^n, \mathbb{R}^m)$; *D, B effective*
- *tolerate error* $\epsilon > 0$ *in decisions*

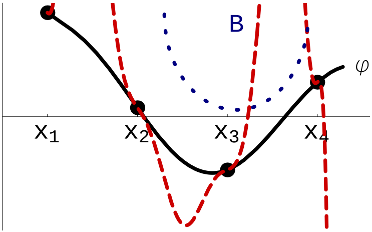


Proposition (Image computation undecidable for…)

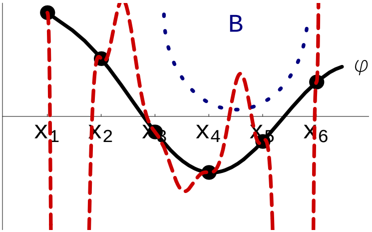
- *arbitrarily effective flow* $\varphi \in C^k(D \subseteq \mathbb{R}^n, \mathbb{R}^m)$*; D, B effective*
- *tolerate error* $\epsilon > 0$ *in decisions*
- φ *smooth polynomial function with* $\mathbb{Q}$*-coefficients*


B
φ
x1
x2
x3

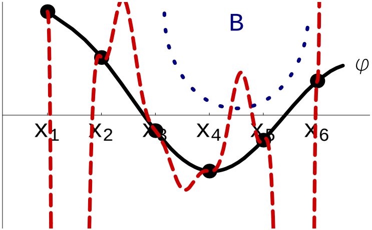
Probabilistic Model Checking



Probabilistic Model Checking



Probabilistic Model Checking

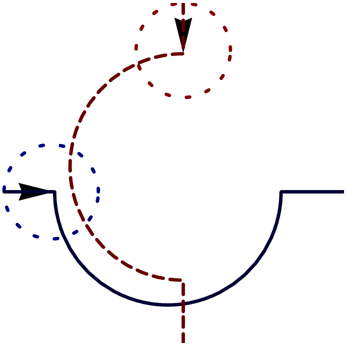


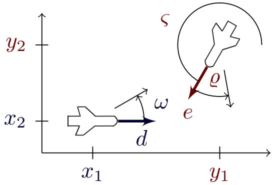
Proposition

- $P(\|\varphi'\|_\infty > b) \to 0$ *as* $b \to \infty$
- φ *evaluated on finite subset* $X = \{x_i\}$ *of open or compact* D

$\Rightarrow$ $P(\textit{decision correct}) \to 1$ *as* $\|d(\cdot, X)\|_\infty \to 0$

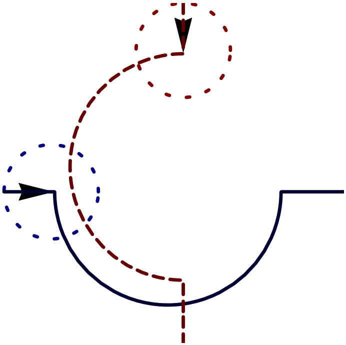
Verification of Nonlinear Hybrid Systems

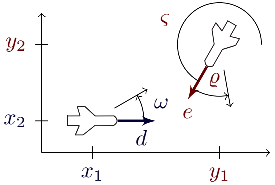




$$\begin{bmatrix} x_1' = -v_1 + v_2 \cos\vartheta + \omega x_2 \\ x_2' = v_2 \sin\vartheta - \omega x_1 \\ \vartheta' = \varrho - \omega \end{bmatrix}$$

Symbolic Verification	Numerical Verification
× constant/nilpotent systems	✓ nonlinear systems
× otherwise "no" solutions	× approximation errors
✓ sound	× sound . . . ?





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

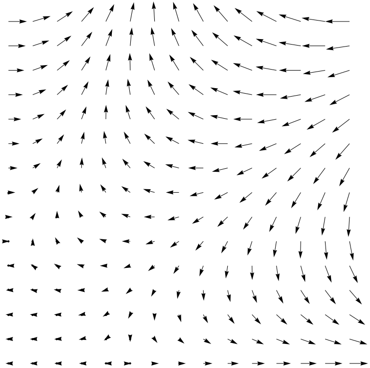
How To Get What We Really Need?

- ✓ nonlinear systems, e.g., curved flight
- ✓ automatic verification
- ✓ sound

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

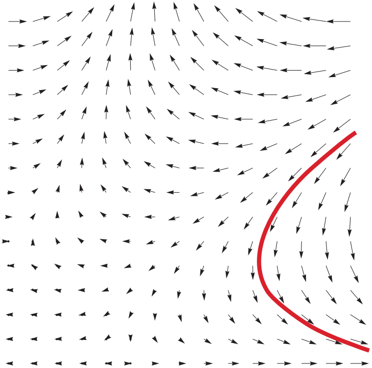
"Property that remains true in the direction of the dynamics"



Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

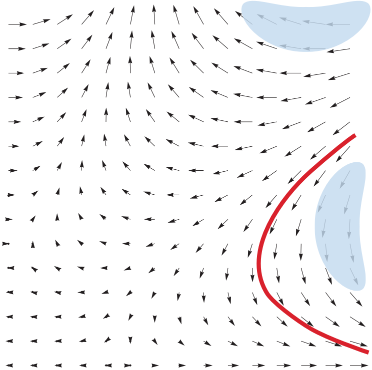
"Property that remains true in the direction of the dynamics"



Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

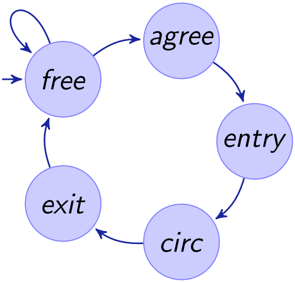
"Property that remains true in the direction of the dynamics"

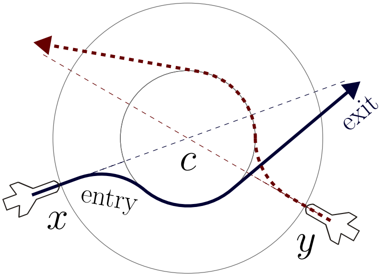


Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

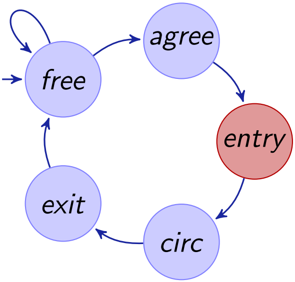




Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

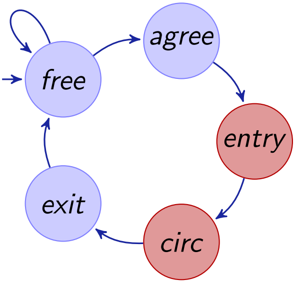


- How to find diff. invariants?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

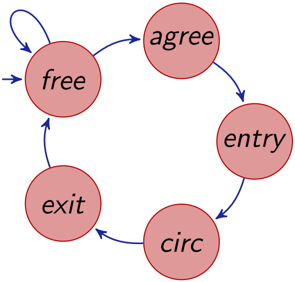


- How to find diff. invariants?
- How do diff. invariants fit together?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

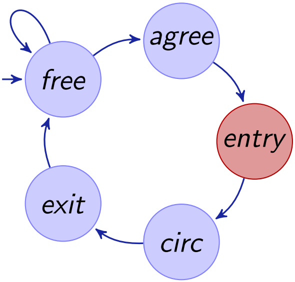


- How to find diff. invariants?
- How do diff. invariants fit together?
- Find all at once? 10000-dim

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

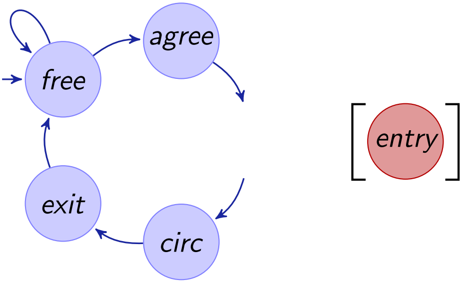


- How to find diff. invariants?
- How do diff. invariants fit together?
- Find local diff. invariants?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

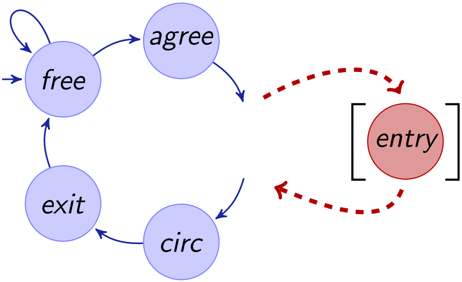


- How to find diff. invariants?
- How do diff. invariants fit together?
- Find local diff. invariants?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

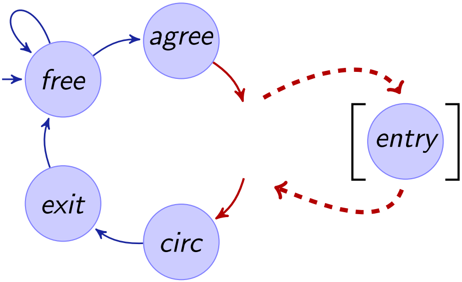


- How to find diff. invariants?
- How do diff. invariants fit together?
- Find local diff. invariants?
- How to put local differential invariants together?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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

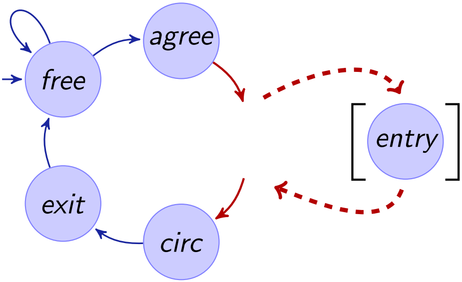


- How to find diff. invariants?
- How do diff. invariants fit together?
- Find local diff. invariants?
- How to put local differential invariants together?
- How do discrete transitions fit?

Idea: Exploit Vector Field of Differential Equations

"Definition" (Differential Invariant)

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



- How to find diff. invariants?
- How do diff. invariants fit together?
- Find local diff. invariants?
- How to put local differential invariants together?
- How do discrete transitions fit?
- What does "fit" really mean?

Outline

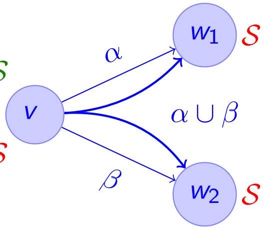
1. CMACS Context
2. Approximation in Model Checking
   - Bounded Flow Approximation
   - Continuous Image Computation
   - Probabilistic Model Checking
   - Differential Invariants
3. Compositional Verification of Hybrid Systems
   - Compositionality in Verification
   - Discrete Induction
   - Differential Induction
4. Computing Differential Invariants by Combining Local Fixedpoints
   - Local Fixedpoints Iteration
   - Global Fixedpoints & Fixedpoint Loop Combinations
5. Collision Avoidance Maneuvers in Air Traffic Control
5. Summary & Plans

Outline

Compositional Verification Basics

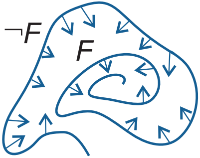
$\alpha \models \mathcal{S}$, $\beta \models \mathcal{S}$

$\alpha \cup \beta \models \mathcal{S}$



Differential Invariants

$$\nabla_{x_1'=f_1(x)\wedge\ldots\wedge x_n'=f_n(x)} F \quad \text{is} \quad \bigwedge_{(b\geq c)\in F} \left(\sum_{i=1}^{n} \frac{\partial b}{\partial x_i} f_i(x) \geq \sum_{i=1}^{n} \frac{\partial c}{\partial x_i} f_i(x) \right)$$



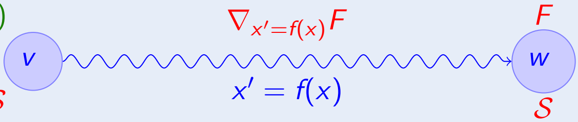
Definition (Differential Invariant F)

F

$(F \rightarrow \mathcal{S})$

$(\nabla_{x'=f(x)} F)$

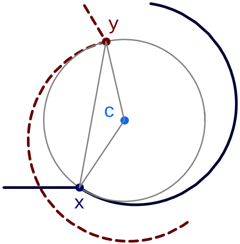
$x' = f(x) \models \mathcal{S}$



Outline

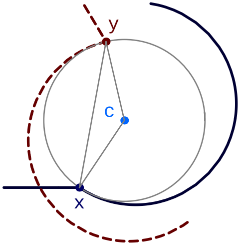
Differential Induction for Roundabout Mode

$$\overline{x_1' = d_1 \wedge d_1' = -\omega d_2 \wedge x_2' = d_2 \wedge d_2' = \omega d_1 \models (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



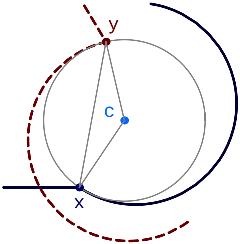
Differential Induction for Roundabout Mode

$$\frac{\frac{\partial\|x-y\|^2}{\partial x_1}x_1' + \frac{\partial\|x-y\|^2}{\partial y_1}y_1' + \frac{\partial\|x-y\|^2}{\partial x_2}x_2' + \frac{\partial\|x-y\|^2}{\partial y_2}y_2' \geq \frac{\partial p^2}{\partial x_1}x_1' \dots}{x_1' = d_1 \land d_1' = -\omega d_2 \land x_2' = d_2 \land d_2' = \omega d_1 \models (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



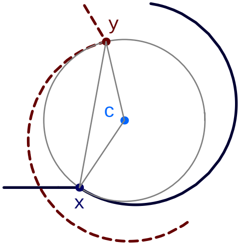
Differential Induction for Roundabout Mode

$$\frac{\frac{\partial\|x-y\|^2}{\partial x_1}x_1' + \frac{\partial\|x-y\|^2}{\partial y_1}y_1' + \frac{\partial\|x-y\|^2}{\partial x_2}x_2' + \frac{\partial\|x-y\|^2}{\partial y_2}y_2' \geq \frac{\partial p^2}{\partial x_1}x_1' \dots}{x_1' = d_1 \land d_1' = -\omega d_2 \land x_2' = d_2 \land d_2' = \omega d_1 \models (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2}$$



Differential Induction for Roundabout Mode

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

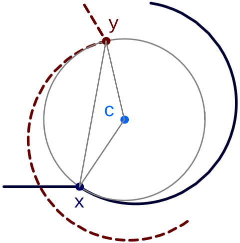


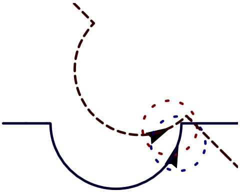
Differential Induction for Roundabout Mode

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

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

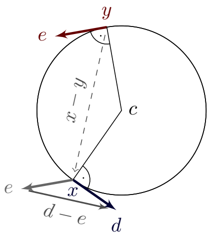
$$x_1' = d_1 \wedge d_1' = -\omega d_2 \wedge x_2' = d_2 \wedge d_2' = \omega d_1 \models (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2$$

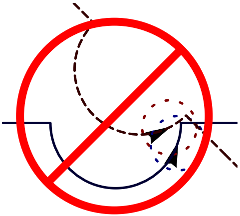




Differential Induction for Roundabout Mode

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





$$\overline{d_1' = -\omega d_2 \land e_1' = -\omega e_2 \land x_2' = d_2 \land d_2' = \omega d_1 .. \models d_1 - e_1 = -\omega(x_2 - y_2)}$$

Differential Induction for Roundabout Mode

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

Proposition (Differential saturation)

F differential invariant of $x' = \theta \land H \models \mathcal{S}$, then

$$x' = \theta \land H \models \mathcal{S} \quad \text{iff} \quad x' = \theta \land H \land F \models \mathcal{S}$$

$$\overline{d_1' = -\omega d_2 \land e_1' = -\omega e_2 \land x_2' = d_2 \land d_2' = \omega d_1 .. \models d_1 - e_1 = -\omega(x_2 - y_2)}$$

Differential Induction for Roundabout Mode

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

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

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

$$x_1' = d_1 \wedge d_1' = -\omega d_2 \wedge x_2' = d_2 \wedge d_2' = \omega d_1 \models (x_1 - y_1)^2 + (x_2 - y_2)^2 \geq p^2$$

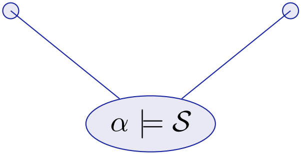
Proposition (Differential saturation)

F differential invariant of $x' = \theta \wedge H \models \mathcal{S}$, then

$$x' = \theta \wedge H \models \mathcal{S} \quad \text{iff} \quad x' = \theta \wedge H \wedge F \models \mathcal{S}$$

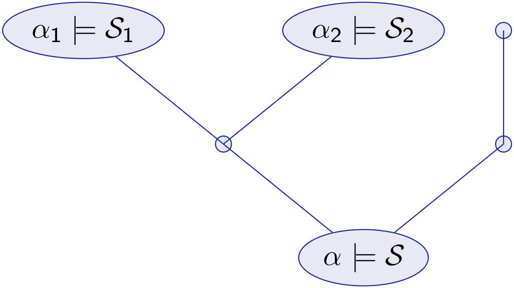
$$\overline{d_1' = -\omega d_2 \wedge e_1' = -\omega e_2 \wedge x_2' = d_2 \wedge d_2' = \omega d_1 .. \models d_1 - e_1 = -\omega(x_2 - y_2)}$$

Differential Invariants as Fixedpoints

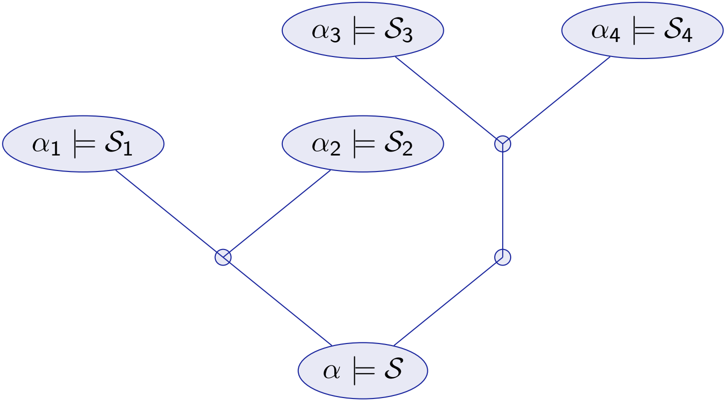


[Clarke'79]

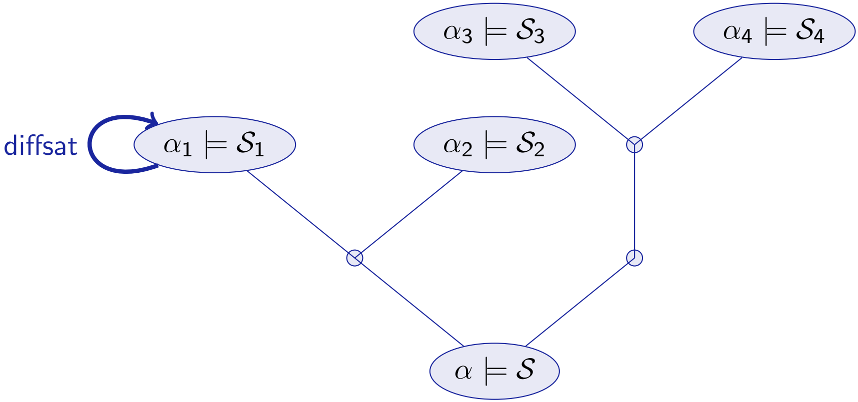
Differential Invariants as Fixedpoints



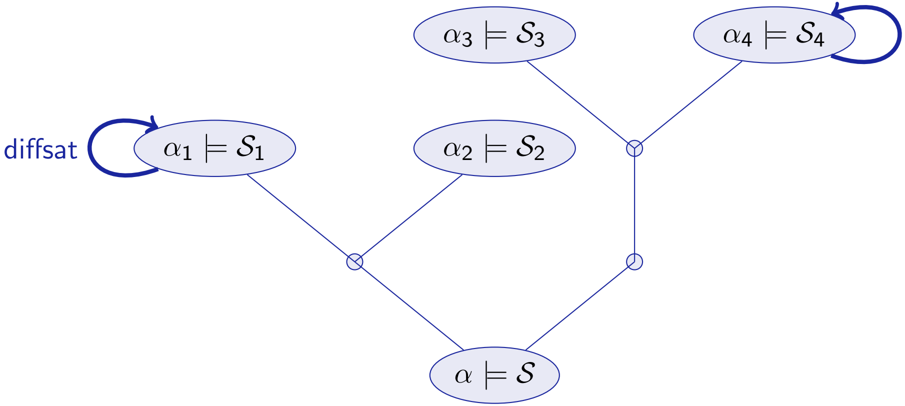
Differential Invariants as Fixedpoints



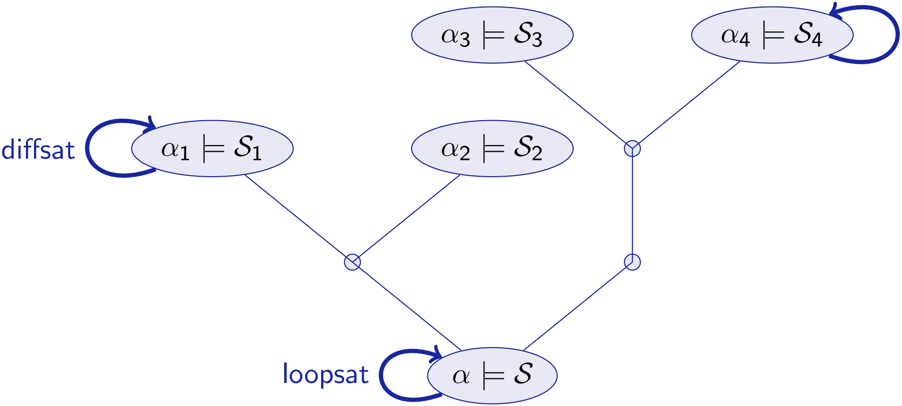
Differential Invariants as Fixedpoints



Differential Invariants as Fixedpoints

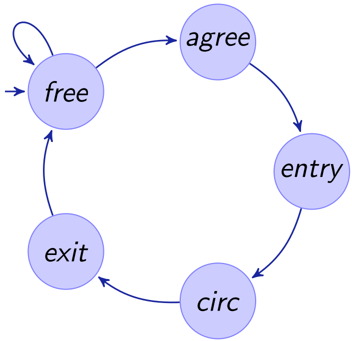


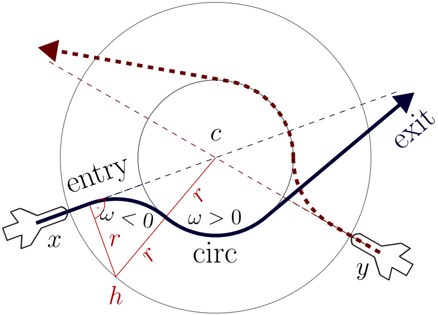
Differential Invariants as Fixedpoints



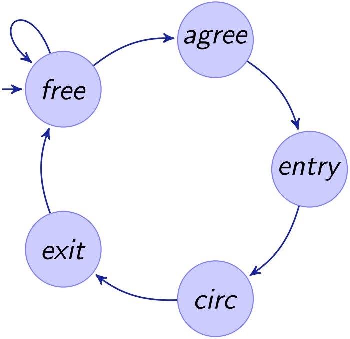
Outline

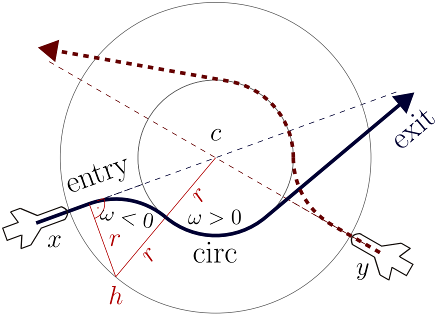
Flyable Roundabout Maneuver: Overview



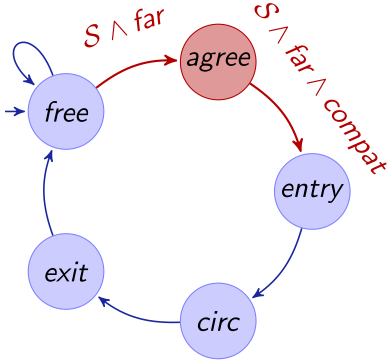


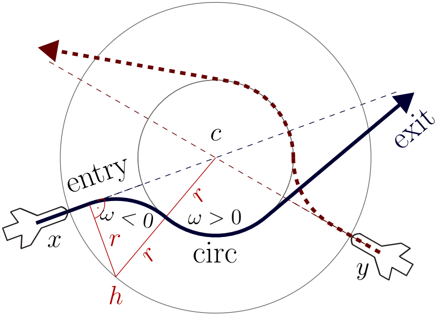
Fixedpoint Iterations for Air Traffic Control



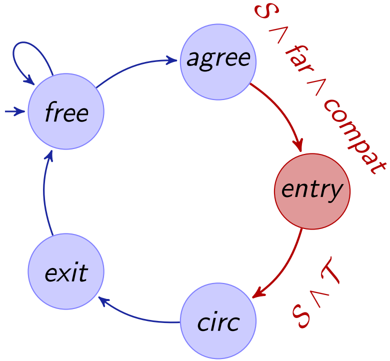


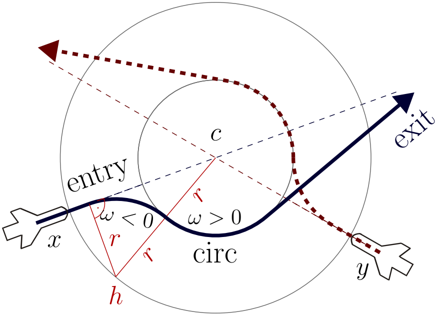
Fixedpoint Iterations for Air Traffic Control



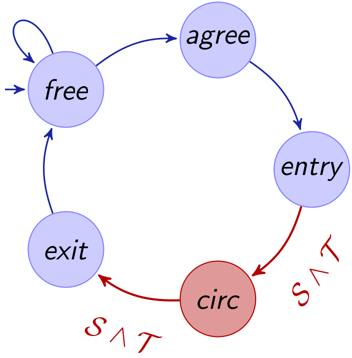


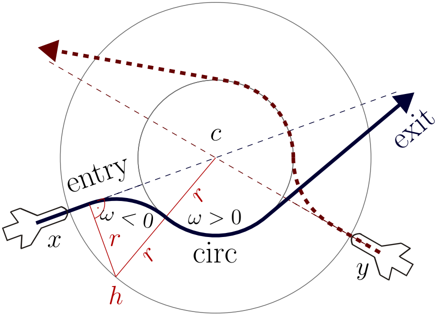
Fixedpoint Iterations for Air Traffic Control

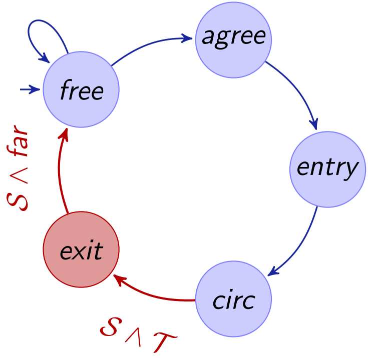


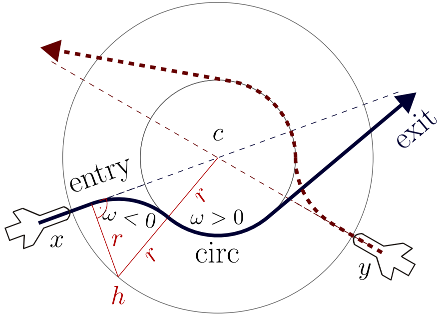


Fixedpoint Iterations for Air Traffic Control

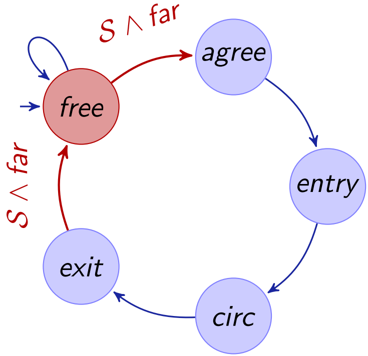


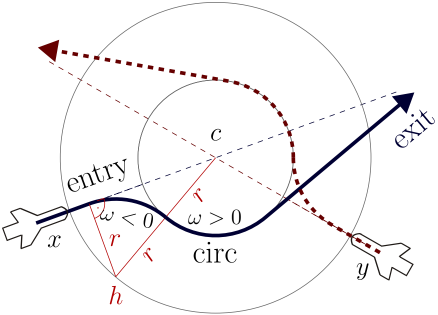



agree
free
entry
exit
circ
$\mathcal{S} \wedge far$
$\mathcal{S} \wedge \mathcal{T}$

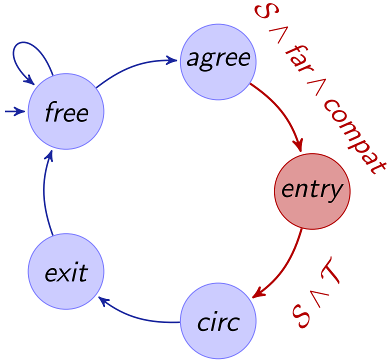

c
entry
exit
r
$\omega < 0$
$\omega > 0$
x
r
r
circ
y
h

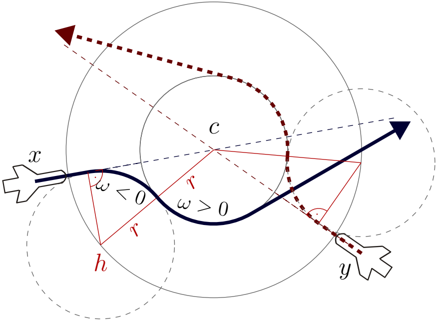
Fixedpoint Iterations for Air Traffic Control



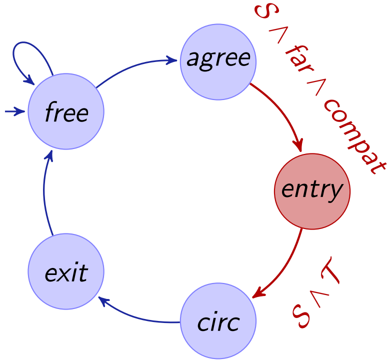


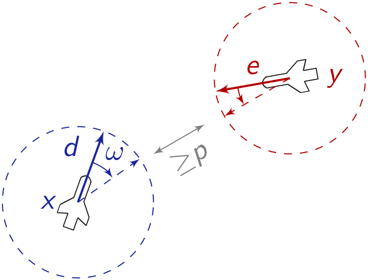
Flyable Roundabout Maneuver: Entry



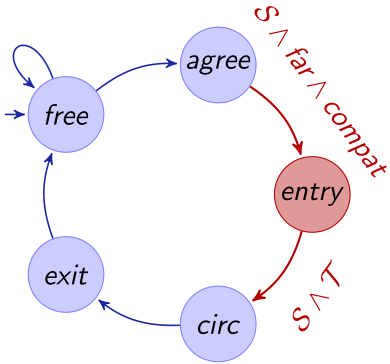


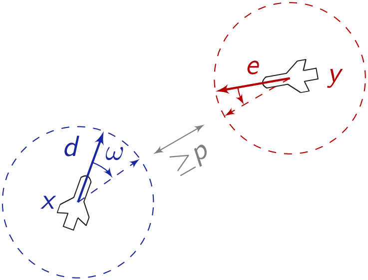
Flyable Roundabout Maneuver: Entry



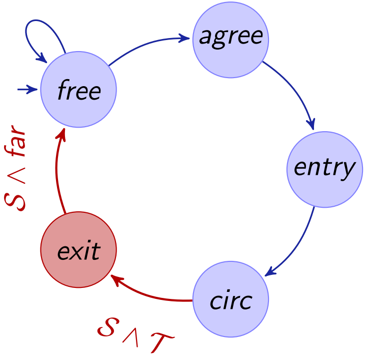


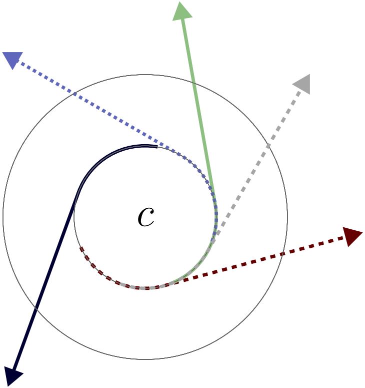
Flyable Roundabout Maneuver: Entry

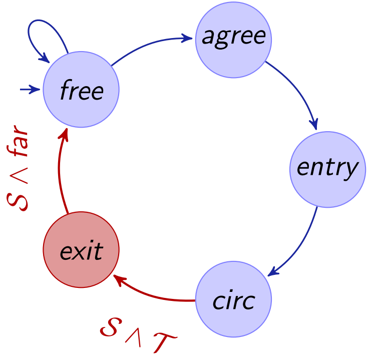


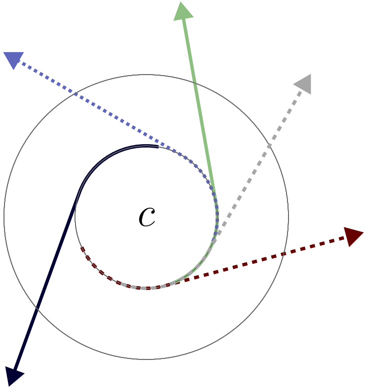


Flyable Roundabout Maneuver: Exit

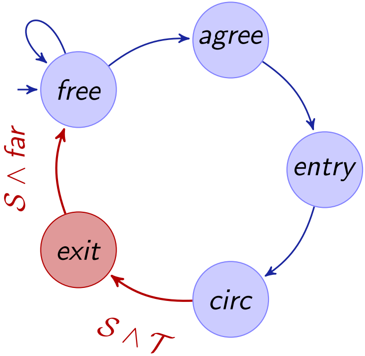


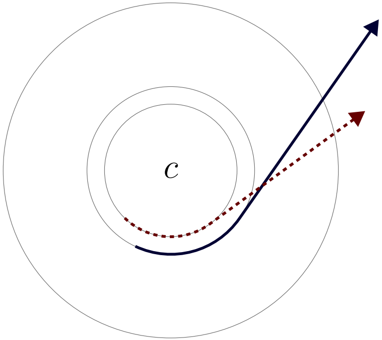



agree
free
entry
exit
circ
$\mathcal{S} \wedge far$
$\mathcal{S} \wedge \mathcal{T}$

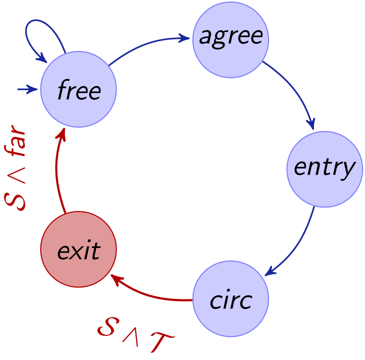

c

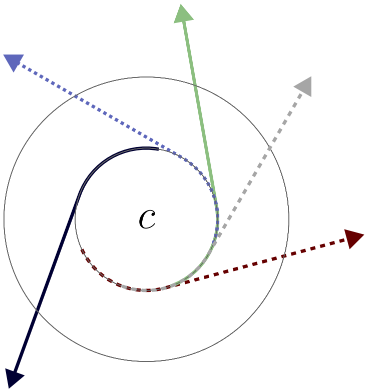
$\mathcal{A}$ Flyable Roundabout Maneuver: Exit



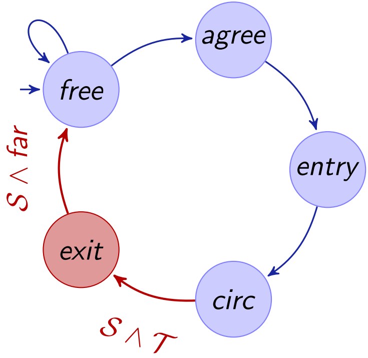


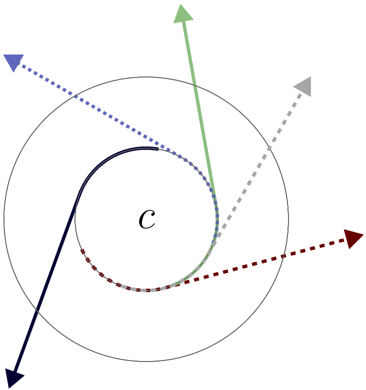
Flyable Roundabout Maneuver: Exit





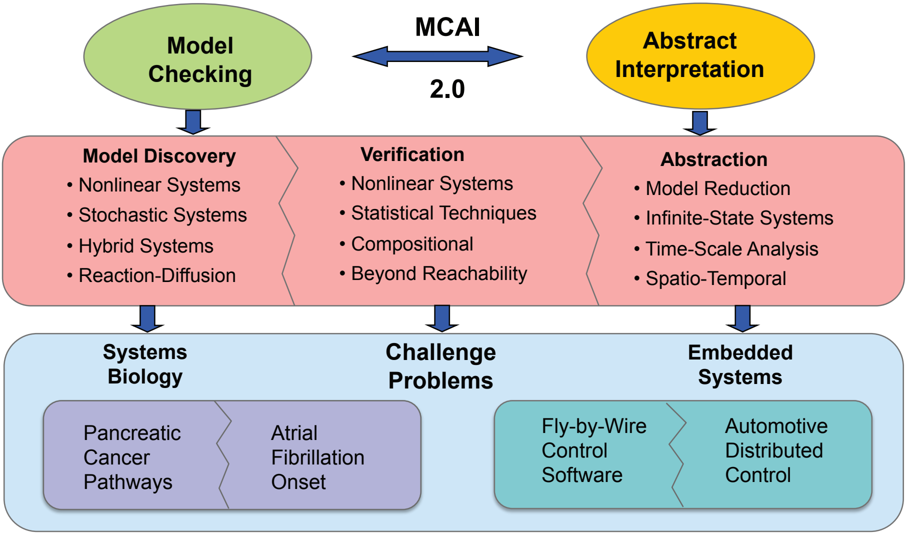
Flyable Roundabout Maneuver: Exit





Outline

Verification: Nonlinear Models & Compositional Models



Plans

- Combining image computation and differential invariants
- Widening for differential invariant fixed points
- Research infrastructure
- Automotive

Integration and Scope

- Verification Aspects
    - Nonlinear models
    - Compositional
    - Beyond reachability
- Challenge Problems
    - Flight domain
    - Automotive control
    - Atrial fibrillation
- Current and envisioned collaborations
    - Ed Clarke (image computation, MC)
    - Patrick Cousot (fixed points, widening, AI)
    - Bruce Krogh (compositionality)
    - Radu Grosu, Flavio Fenton, ... (wave-front curvatures and collisions in AFib)
    - Paolo Zuliani, Steve Marcus, ... (see statistical model checking talk later today)
    - ...