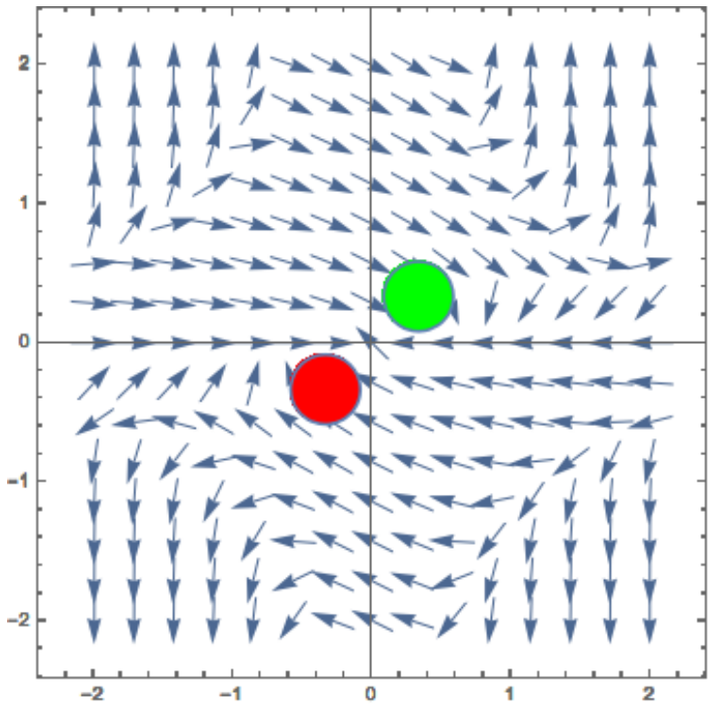
Symbolic Semialgebraic Decomposition for Polynomial ~~Hybrid~~ Dynamical Systems

Sergio Mover
Ecole Polytechnique

Joint work with:
Alessandro Cimatti, FBK, Italy
Alberto Griggio, FBK, Italy
Stefano Tonetta, FBK, Italy
Ahmed Irfan, Stanford University

Unbounded verification for polynomial systems

$$\dot{X} = f(X)$$



$$\dot{x} = -x + 2y + x^2y + x^4y^5$$

$$\dot{y} = -y - x^4y^6 + x^8y^9$$

Polynomial dynamic

$$I := \left(x - \frac{1}{3}\right)^2 + \left(y - \frac{1}{3}\right)^2 < \frac{1}{16}$$

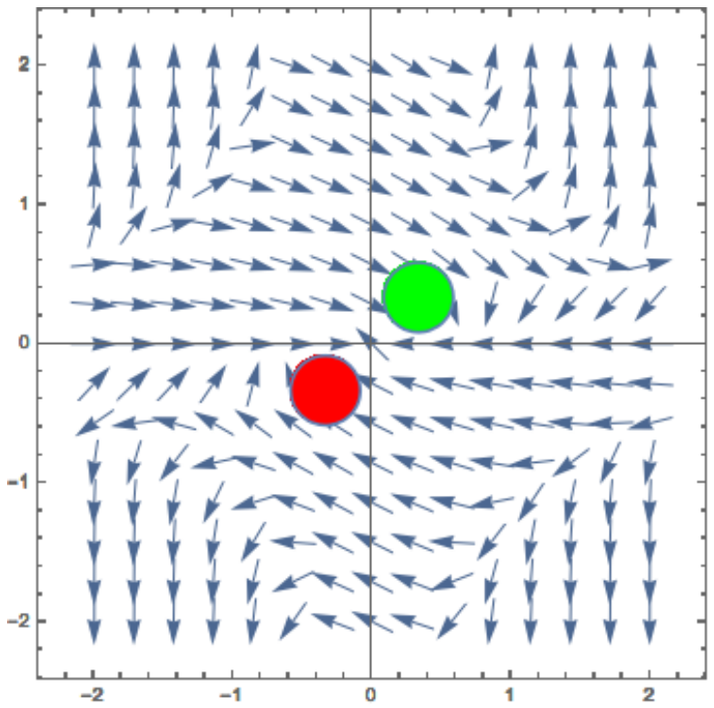
Initial states

$$\psi := \left(-x - \frac{1}{3}\right)^2 + \left(-y - \frac{1}{3}\right)^2 \geq \frac{1}{16}$$

Safe states

Unbounded verification for polynomial systems

$$\dot{X} = f(X)$$



$$\dot{x} = -x + 2y + x^2y + x^4y^5$$
$$\dot{y} = -y - x^4y^6 + x^8y^9$$

Polynomial dynamic

$$I := \left(x - \frac{1}{3}\right)^2 + \left(y - \frac{1}{3}\right)^2 < \frac{1}{16}$$

Initial states

$$\psi := \left(-x - \frac{1}{3}\right)^2 + \left(-y - \frac{1}{3}\right)^2 \geq \frac{1}{16}$$

Safe states

$$H \wedge \psi \rightarrow \sigma$$

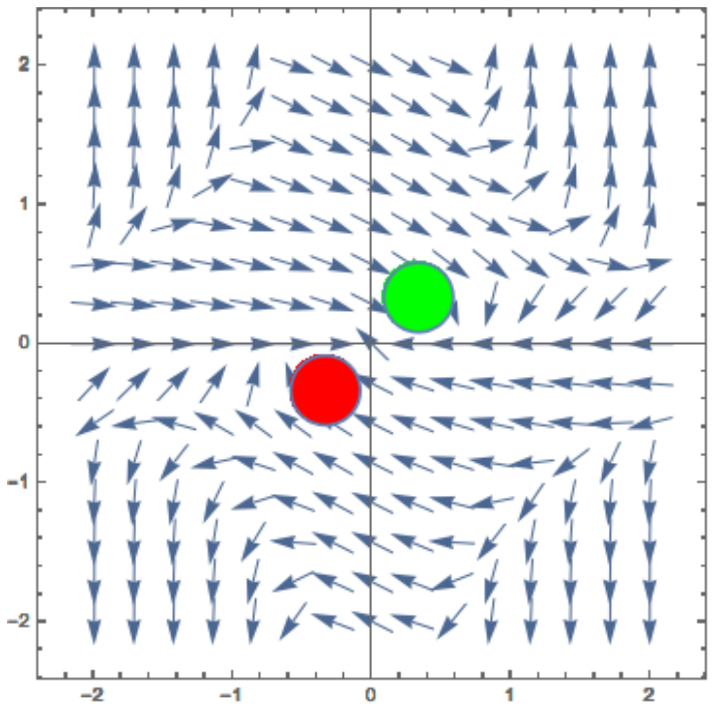
Differential invariant

$$\sigma \rightarrow [\dot{X} = f(X)\ \&\ H]\ \sigma$$
$$\sigma \rightarrow \psi$$

Problem: find an invariant sufficient to prove safety

Unbounded verification for polynomial systems

$$\dot{X} = f(X)$$



$$\dot{x} = -x + 2y + x^2y + x^4y^5$$
$$\dot{y} = -y - x^4y^6 + x^8y^9$$

Polynomial dynamic

$$I := \left(x - \frac{1}{3}\right)^2 + \left(y - \frac{1}{3}\right)^2 < \frac{1}{16}$$

Initial states

$$\psi := \left(-x - \frac{1}{3}\right)^2 + \left(-y - \frac{1}{3}\right)^2 \geq \frac{1}{16}$$

Safe states

$$H \wedge \psi \rightarrow \sigma$$

Differential invariant

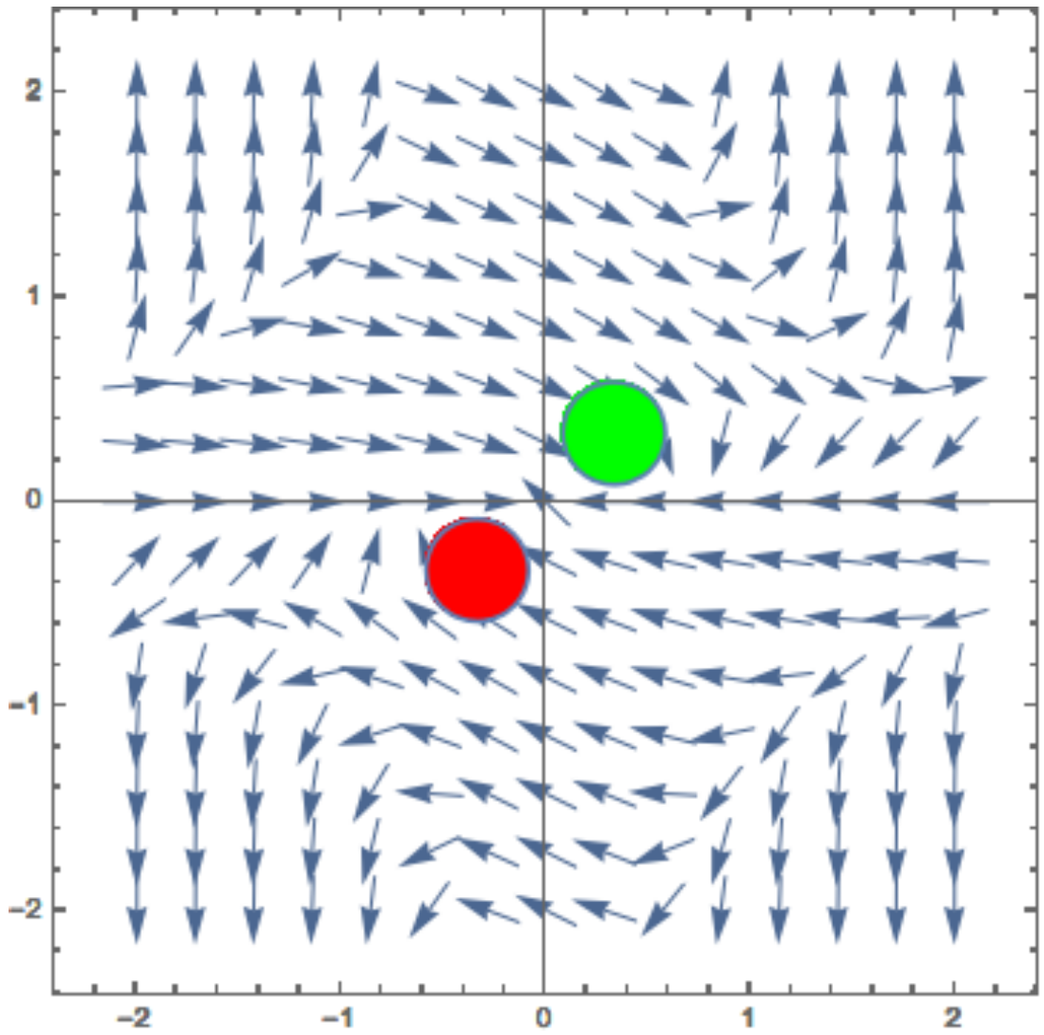
$$\sigma \rightarrow [\dot{X} = f(X)\ \&\ H]\ \sigma$$
$$\sigma \rightarrow \psi$$

Problem: find an invariant sufficient to prove safety

How can we find the differential invariant?

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$



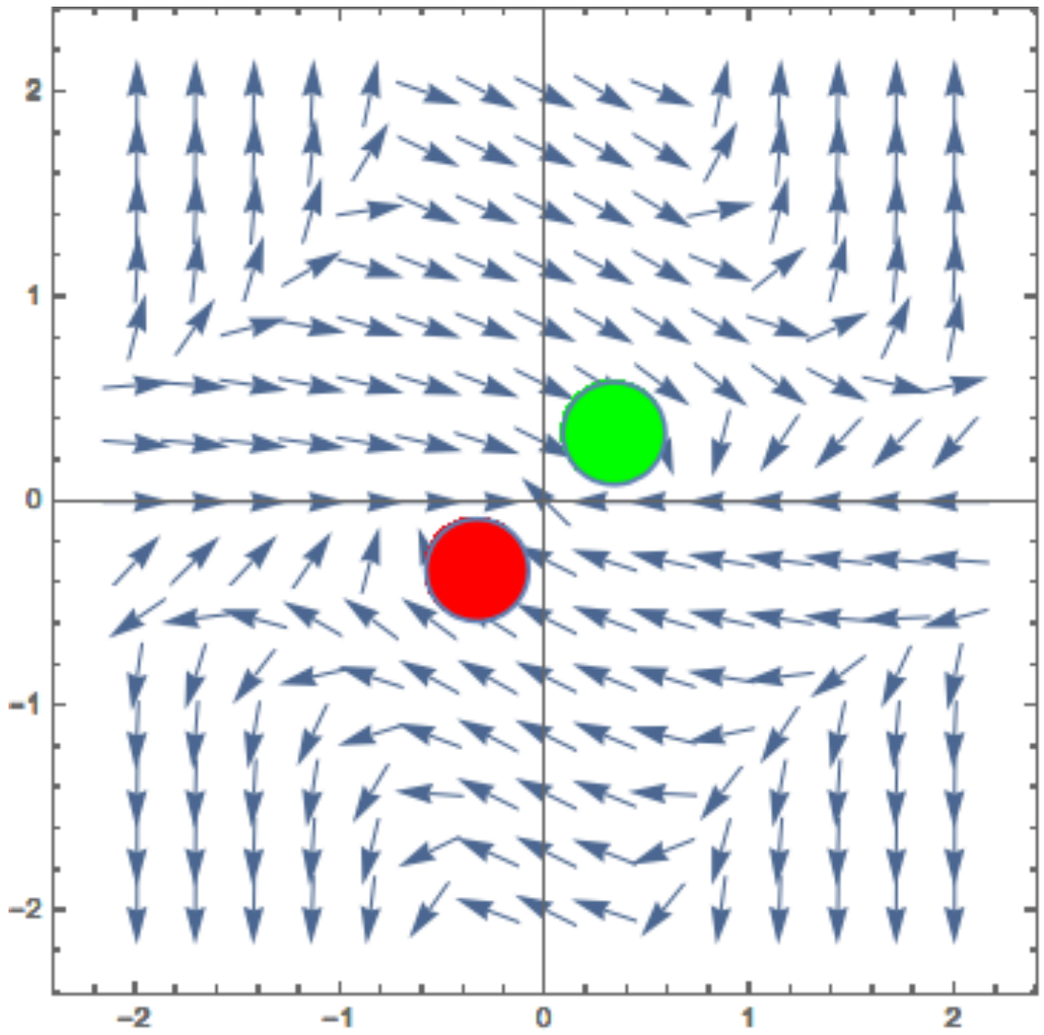
Defines 3^A abstract states…

Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction



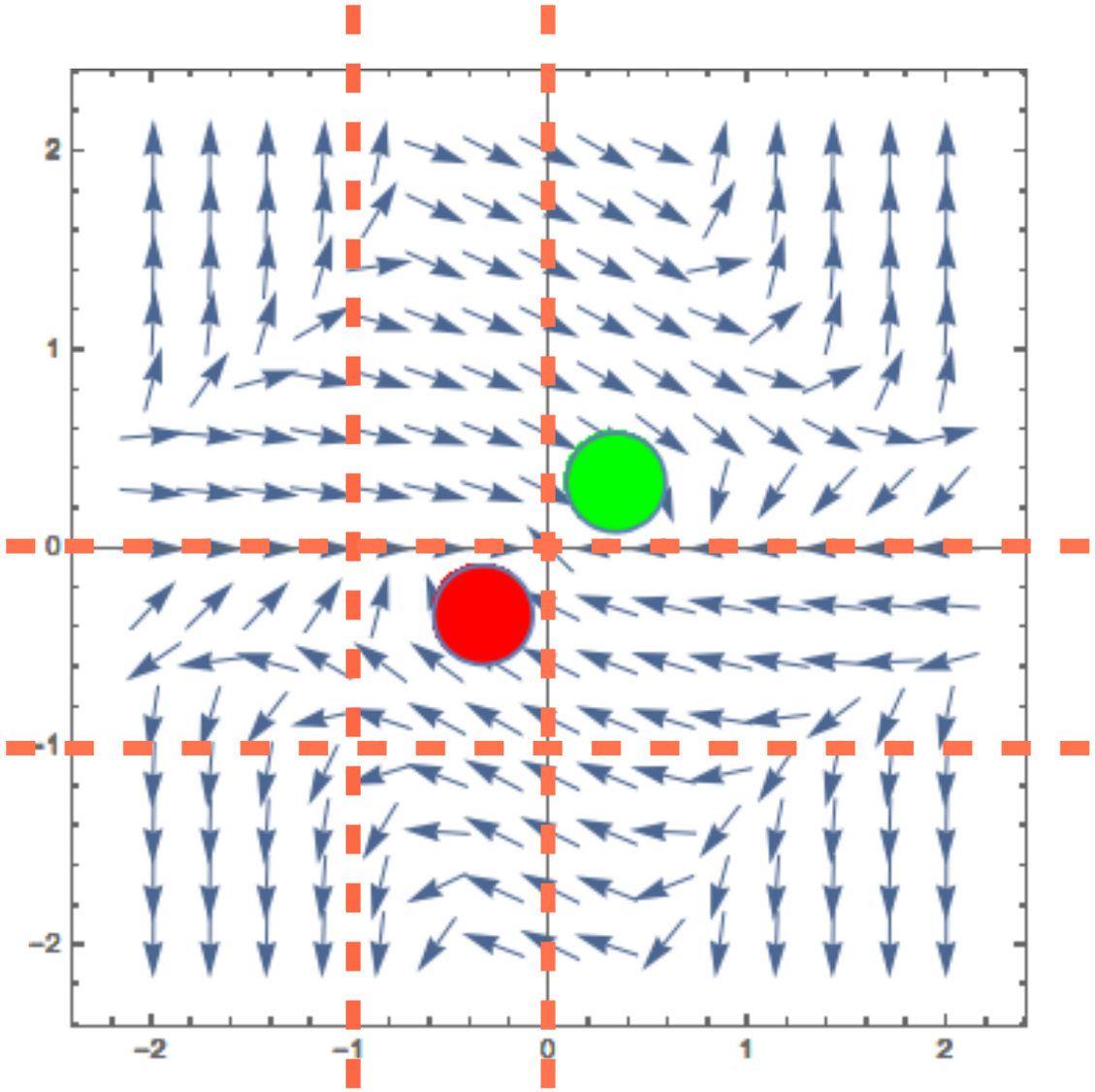
Defines 3^A abstract states…

Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction



Defines 3^A abstract states…

Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

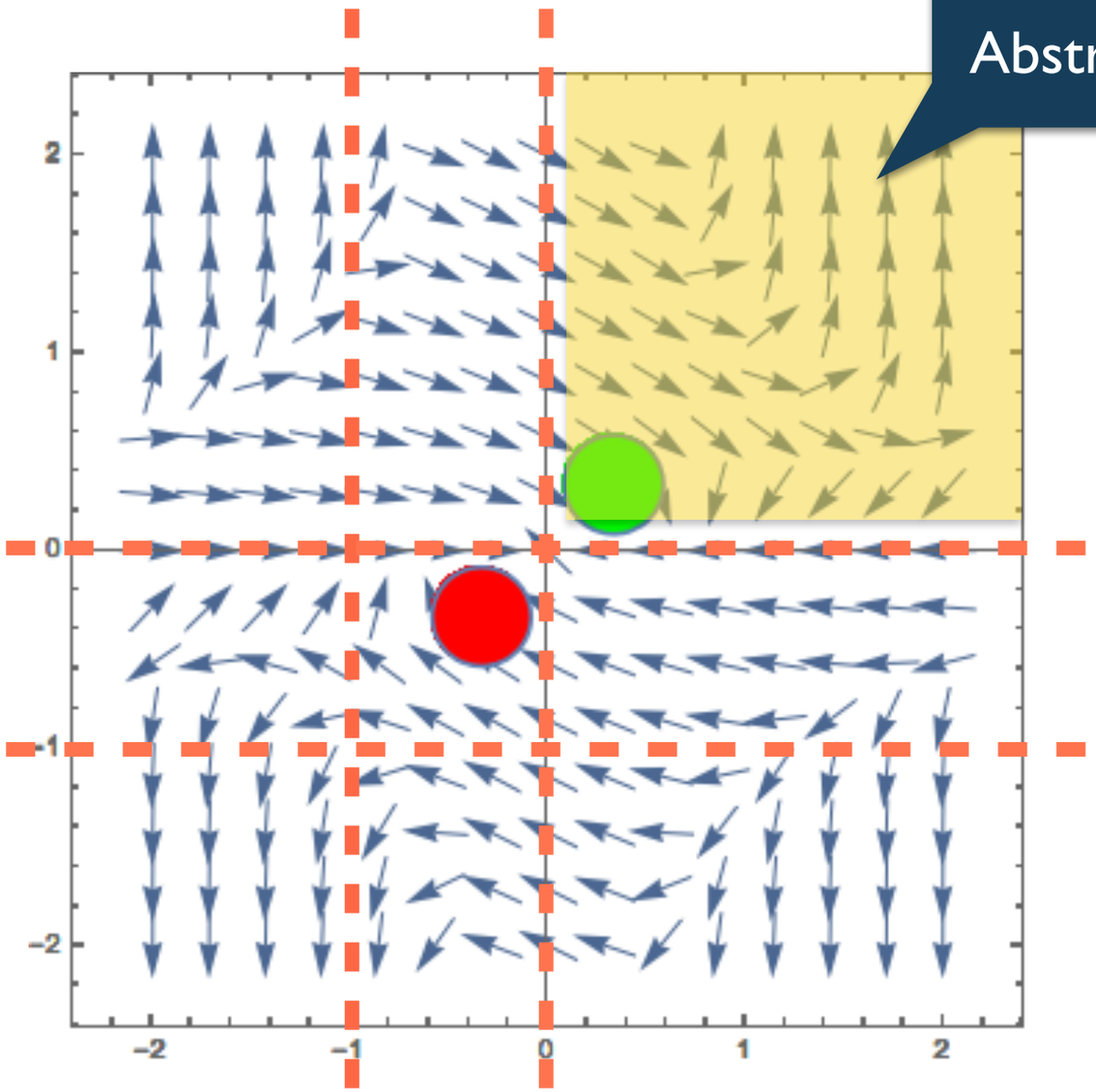
Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state

$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$



Defines 3^A abstract states…

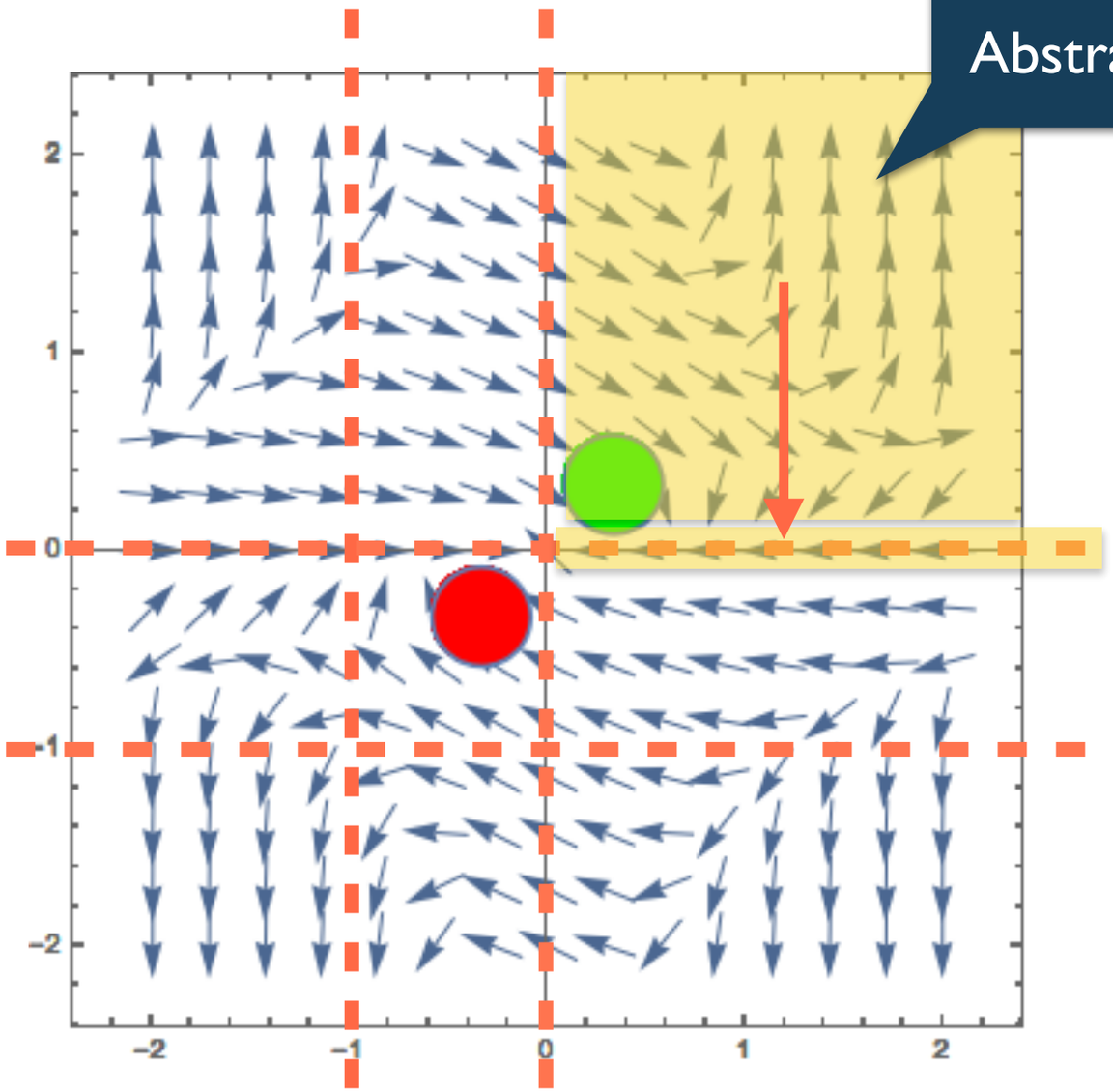
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state



$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$

$$0 < x \wedge 0 = y \wedge 0 < x+1 \wedge 0 < y+1$$

Check the existence of a transition between abstract states

Defines 3^A abstract states…

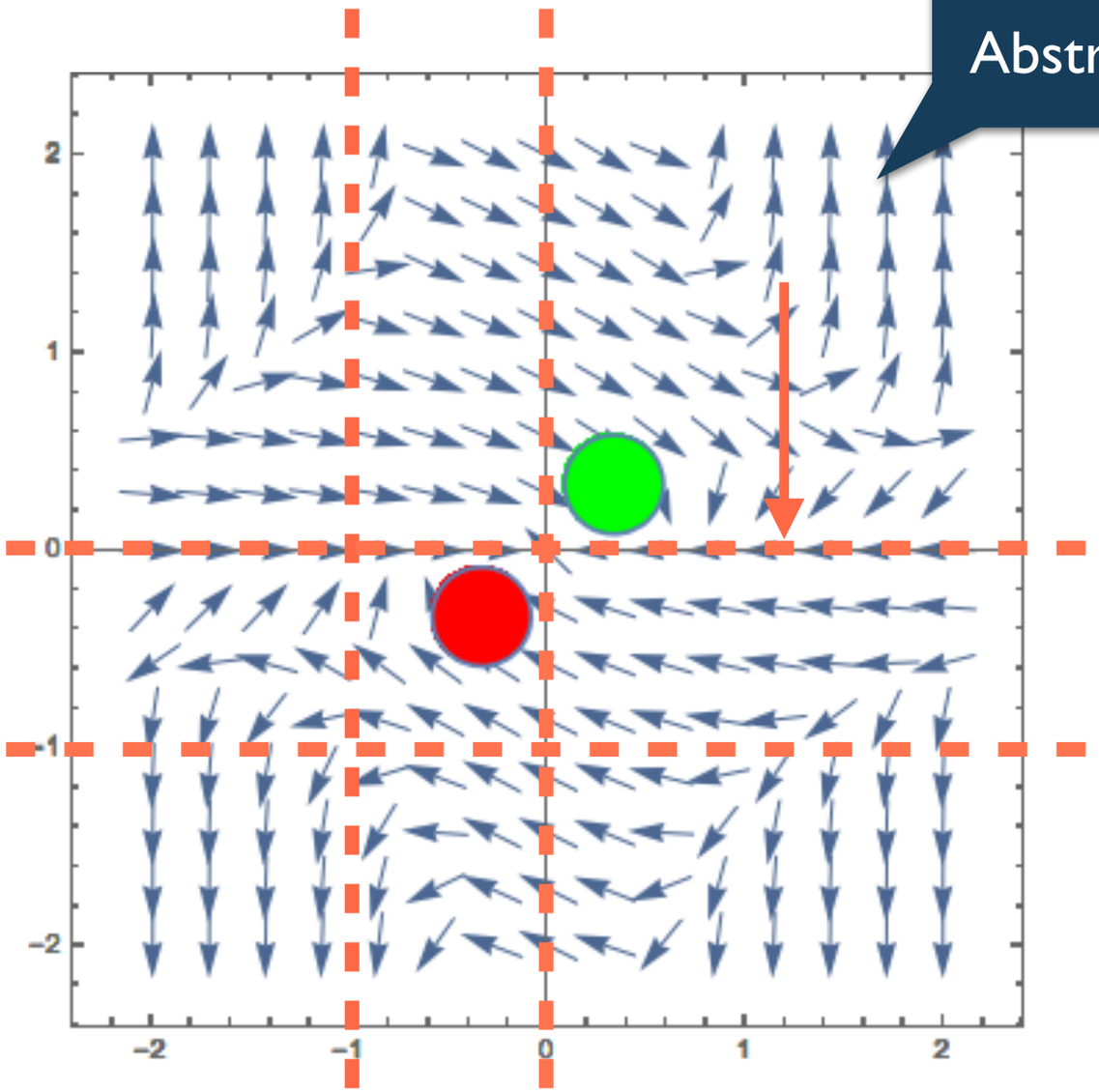
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state



$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$

$$0 < x \wedge 0 = y \wedge 0 < x+1 \wedge 0 < y+1$$

Check the existence of a transition between abstract states

Defines 3^A abstract states…

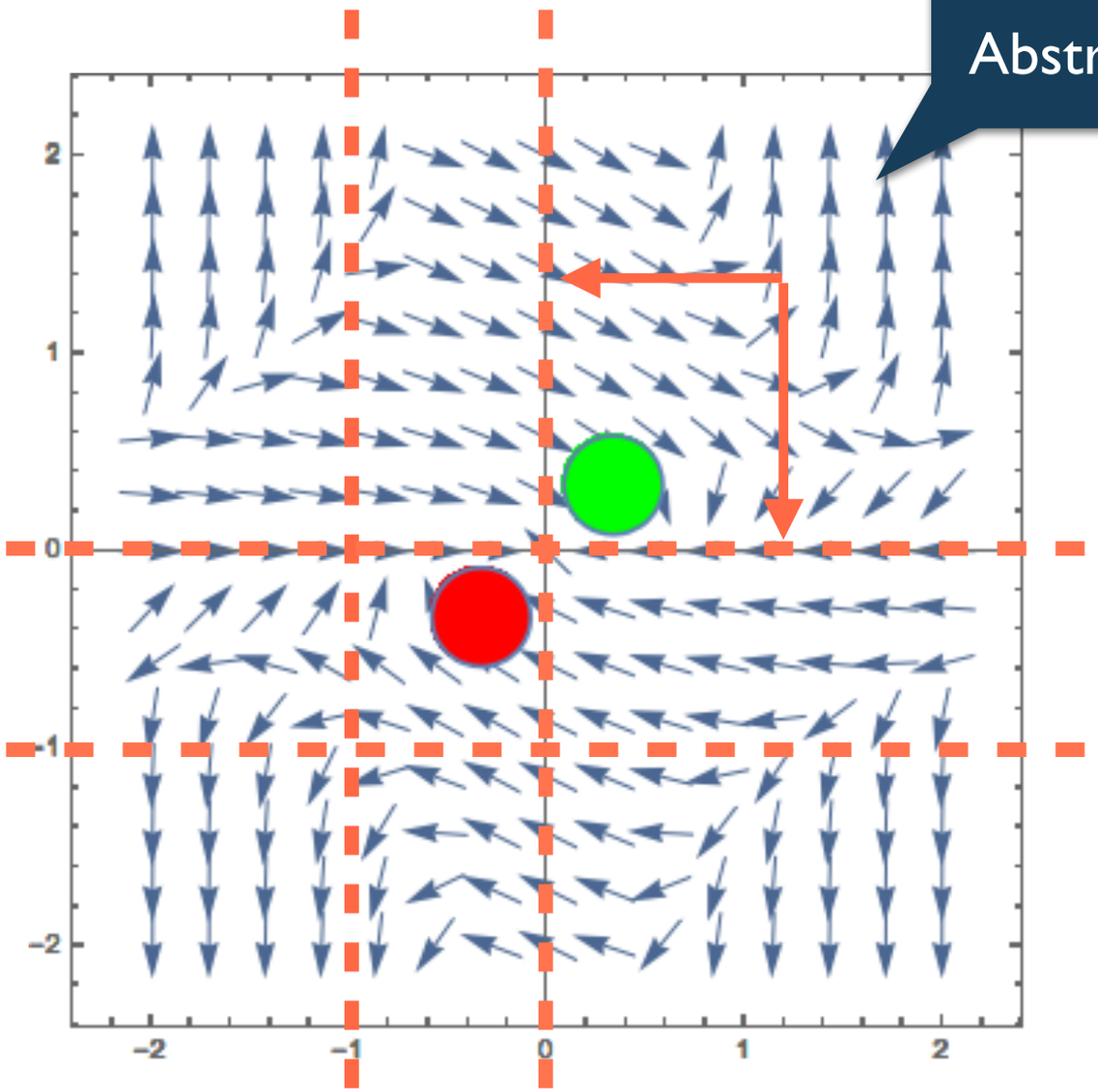
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state



$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$

$$0 < x \wedge 0 = y \wedge 0 < x+1 \wedge 0 < y+1$$

Check the existence of a transition between abstract states

Defines 3^A abstract states…

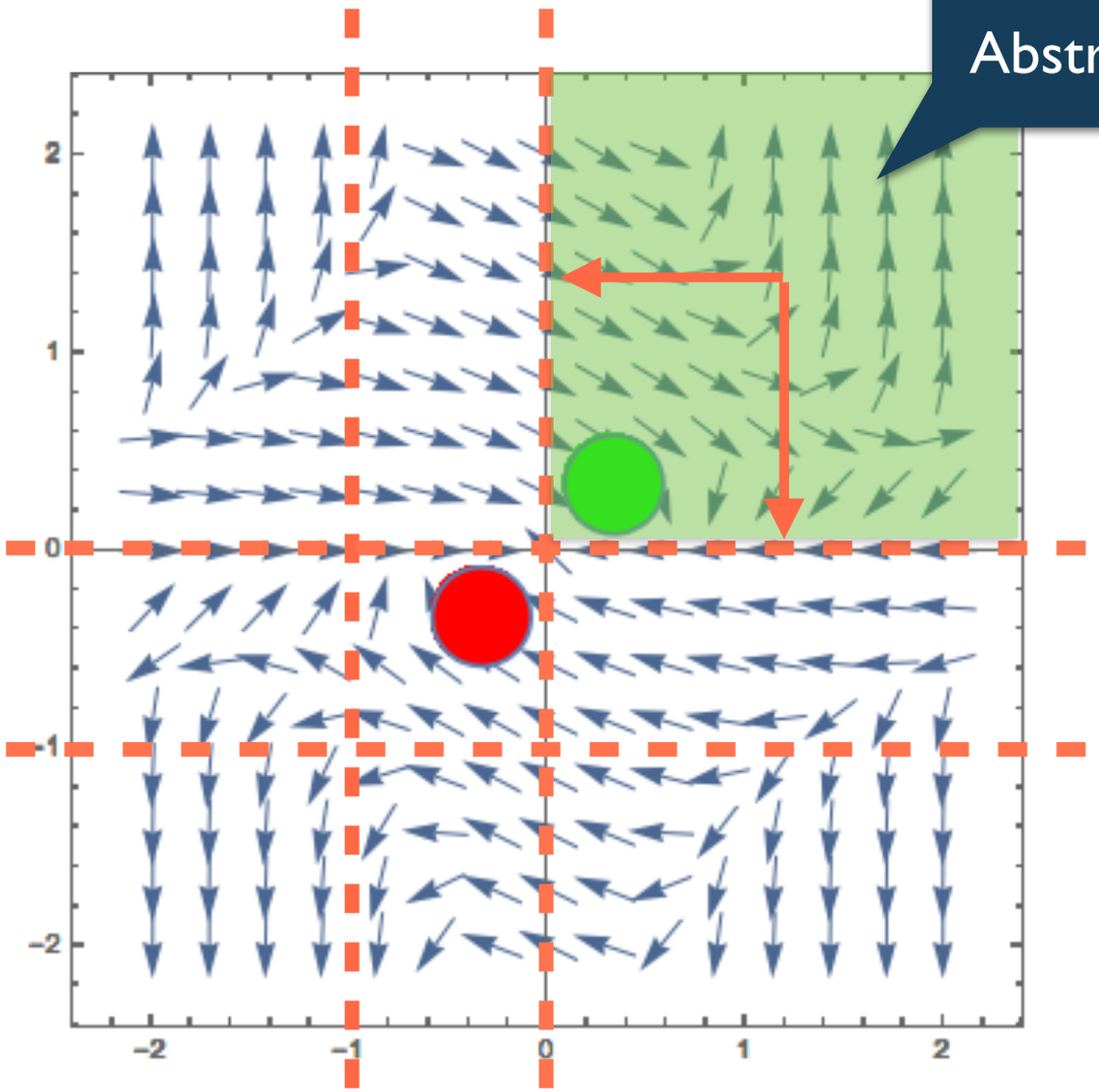
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state



$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$

$$0 < x \wedge 0 = y \wedge 0 < x+1 \wedge 0 < y+1$$

Check the existence of a transition between abstract states

Defines 3^A abstract states…

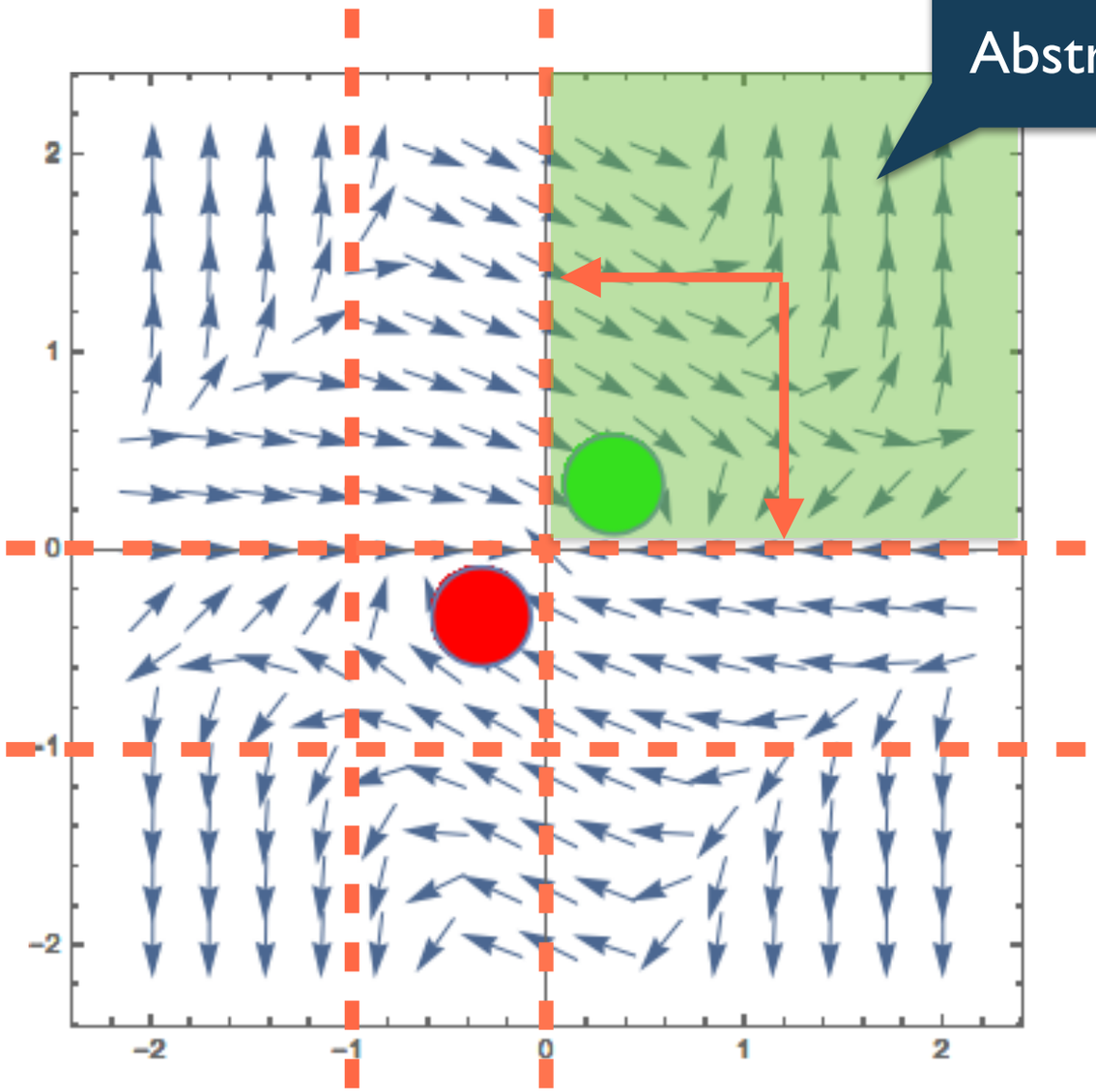
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Decomposing the state space: Semialgebraic Decomposition

$$A = \{x, y, x+1, y+1\}$$

Polynomials for the abstraction

Abstract state



$$0 < x \wedge 0 < y \wedge 0 < x+1 \wedge 0 < y+1$$

$$0 < x \wedge 0 = y \wedge 0 < x+1 \wedge 0 < y+1$$

Check the existence of a transition between abstract states

Defines 3^A abstract states

This is a predicate abstraction

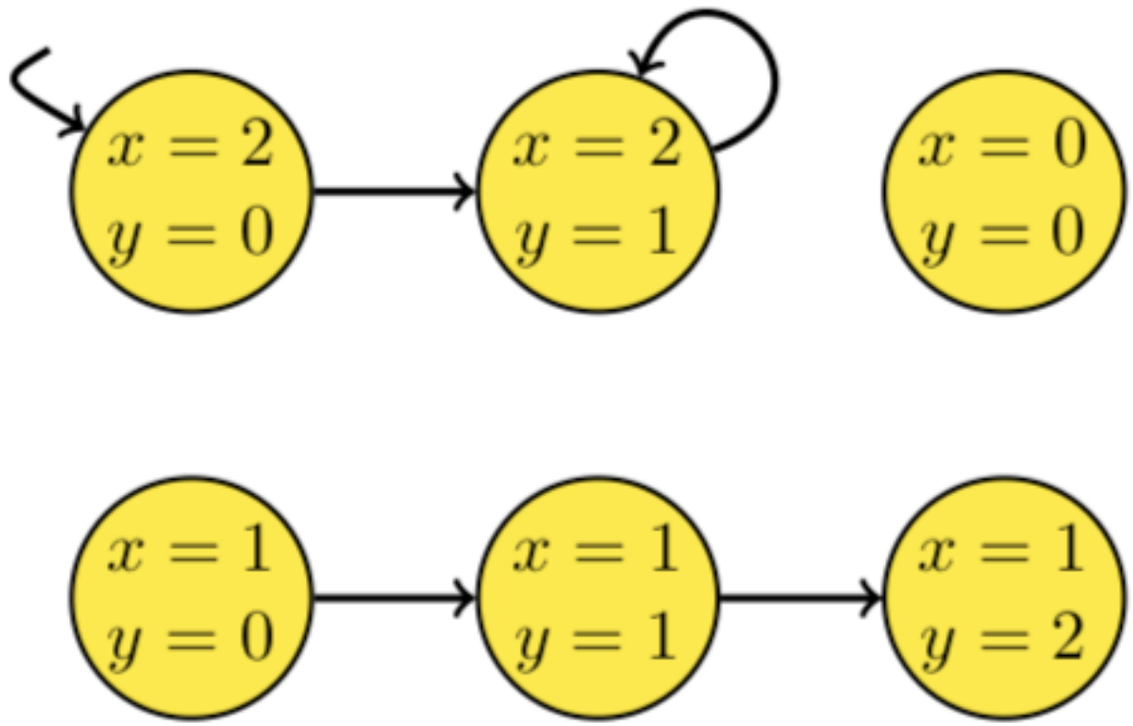
Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016

Predicate abstraction for discrete systems

$$S = \langle X, I(X), T(X, X') \rangle$$

Discrete transition system

$$\begin{aligned} X :=& \{x, y\} \\ I(X) :=& x = 2 \wedge y = 0 \\ T(X, X') :=& (x = 2 \rightarrow (x' = 2 \wedge y < 2)) \wedge \\ & (x = 1 \rightarrow x' = 1) \wedge \\ & y' = y + 1 \wedge y < 3 \end{aligned}$$



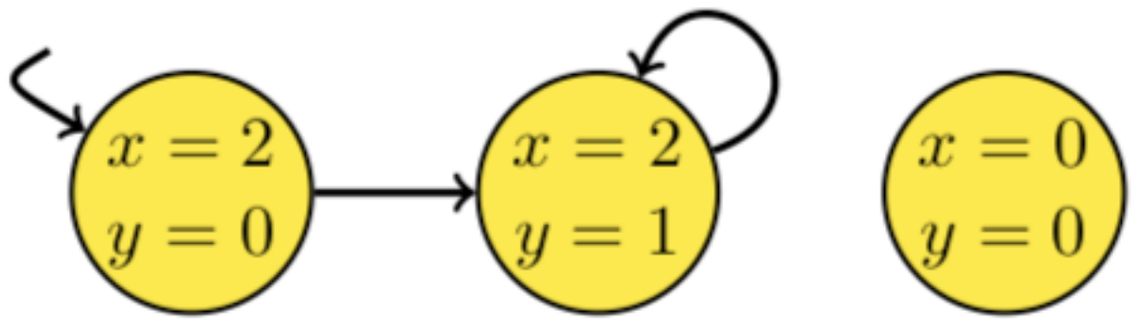
Example from Kroening, SSFT12

Predicate abstraction for discrete systems

$$S = \langle X, I(X), T(X, X') \rangle$$

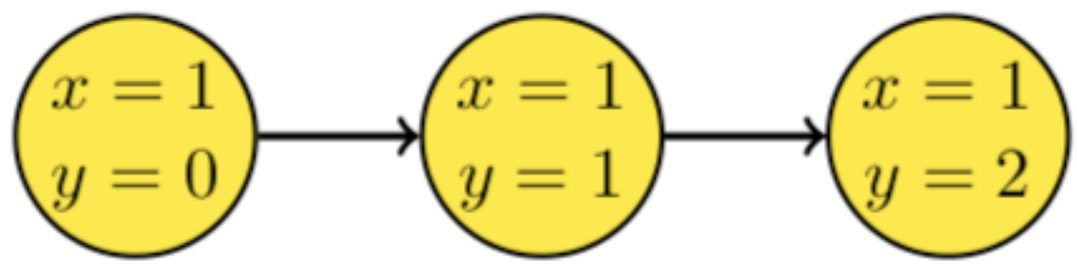
Discrete transition system

$$\begin{aligned} X :=& \{x, y\} \\ I(X) :=& x = 2 \wedge y = 0 \\ T(X, X') :=& (x = 2 \rightarrow (x' = 2 \wedge y < 2)) \wedge \\ & (x = 1 \rightarrow x' = 1) \wedge \\ & y' = y + 1 \wedge y < 3 \end{aligned}$$



Set of predicates

$$p1 := x > y$$
$$p2 := y = 0$$



Example from Kroening, SSFT12

Predicate abstraction for discrete systems

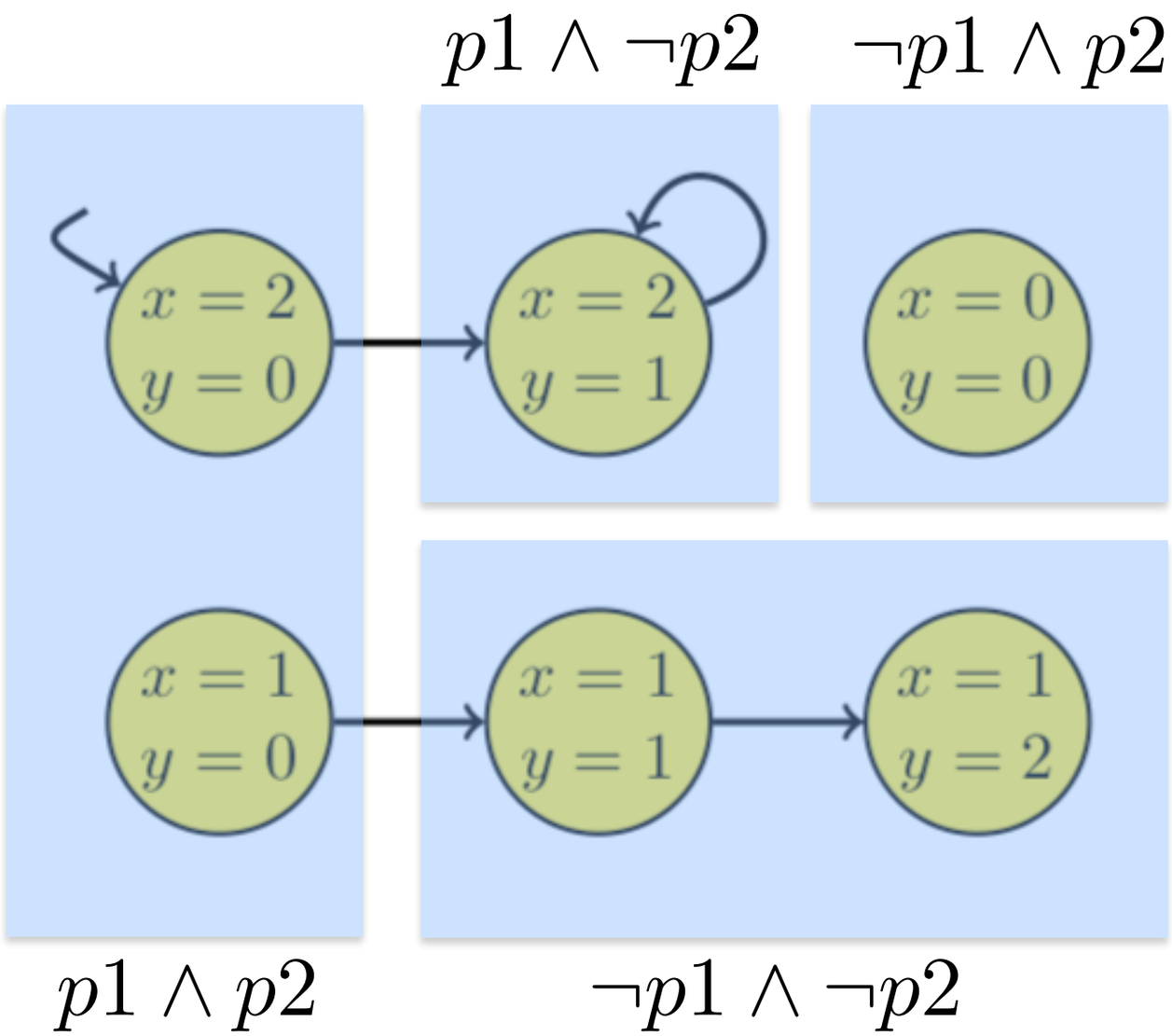
$$S = \langle X, I(X), T(X, X') \rangle$$

Discrete transition system

$$\begin{aligned} X :=& \{x, y\} \\ I(X) :=& x = 2 \wedge y = 0 \\ T(X, X') :=& (x = 2 \rightarrow (x' = 2 \wedge y < 2)) \wedge \\ & (x = 1 \rightarrow x' = 1) \wedge \\ & y' = y + 1 \wedge y < 3 \end{aligned}$$

Set of predicates

$$\begin{aligned} p1 &:= x > y \\ p2 &:= y = 0 \end{aligned}$$



Example from Kroening, SSFT12

Symbolic computation of predicate abstraction

$$S = \langle X, I(X), T(X, X') \rangle$$

$$P = \{a_1(X) < 0, a_2(X) = 0, \ldots\}$$

Symbolic computation of predicate abstraction

$$S = \langle X, I(X), T(X, X') \rangle$$

$$P = \{a_1(X) < 0, a_2(X) = 0, \ldots\}$$

Discrete transition system

Symbolic computation of predicate abstraction

$$S = \langle X, I(X), T(X, X') \rangle$$

Discrete transition system

$$P = \{a_1(X) < 0, a_2(X) = 0, \ldots\}$$

Set of predicates

Symbolic computation of predicate abstraction

$$S = \langle X, I(X), T(X, X') \rangle$$

Discrete transition system

$$P = \{a_1(X) < 0, a_2(X) = 0, \ldots\}$$

Set of predicates

Compute the predicate abstraction

$$S_P = \langle V_P, I_P, T_P \rangle$$

$$V_P = \{v_p \in \mathbb{B} \mid p \in P\}$$

$$I_P = \exists X.(I(X) \wedge \bigwedge_{p \in P} (v_p \leftrightarrow p(X)))$$

$$T_P = \exists X, X'.(T(X, X') \wedge \bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')))$$

Lahiri et al CAV 2006

Symbolic computation of predicate abstraction

$$S = \langle X, I(X), T(X, X') \rangle$$

Discrete transition system

$$P = \{a_1(X) < 0, a_2(X) = 0, \ldots\}$$

Set of predicates

Compute the predicate abstraction

$$S_P = \langle V_P, I_P, T_P \rangle$$

$$V_P = \{v_p \in \mathbb{B} \mid p \in P\}$$

$$I_P = \exists X.(I(X) \wedge \bigwedge_{p \in P} (v_p \leftrightarrow p(X)))$$

$$T_P = \exists X, X'.(T(X, X') \wedge \bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')))$$

Quantifier elimination (using a SMT solver…)

Lahiri et al CAV 2006

Symbolic computation of predicate abstraction

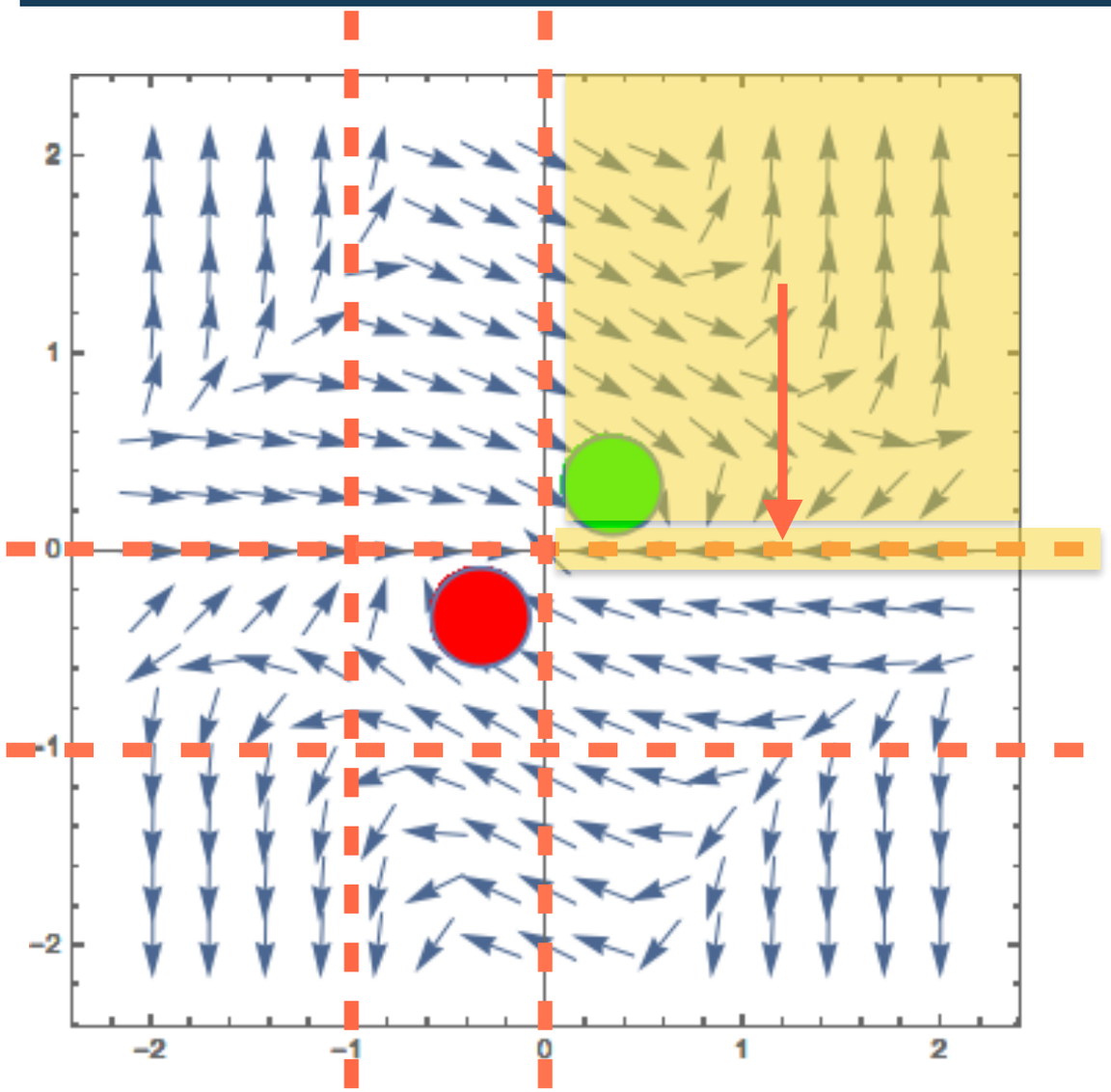
- Efficient handling of the exponential blow-up in computing the abstraction and fully automatic verification algorithms:
  - efficient quantification — Lahiri et al CAV 2006
  - automatic abstraction refinement — Henzinger et al POPL 2004
  - implicit predicate abstraction and IC3 — Tonetta FM 2009; Cimatti et al TACAS 2014
- Issues with semialgebraic decomposition
  - Explicit computation of reachable states
  - Computes reachable states vs. finding a sufficient invariant
  - What does happen when we have hybrid systems?

Symbolic computation of predicate abstraction

- Efficient handling of the exponential blow-up in computing the abstraction and fully automatic verification algorithms:
  - efficient quantification — Lahiri et al CAV 2006
  - automatic abstraction refinement — Henzinger et al POPL 2004
  - implicit predicate abstraction and IC3 — Tonetta FM 2009; Cimatti et al TACAS 2014
- Issues with semialgebraic decomposition
  - Explicit computation of reachable states
  - Computes reachable states vs. finding a sufficient invariant
  - What does happen when we have hybrid systems?

Can we apply symbolic techniques to compute the semialgebraic decomposition?

Main challenges

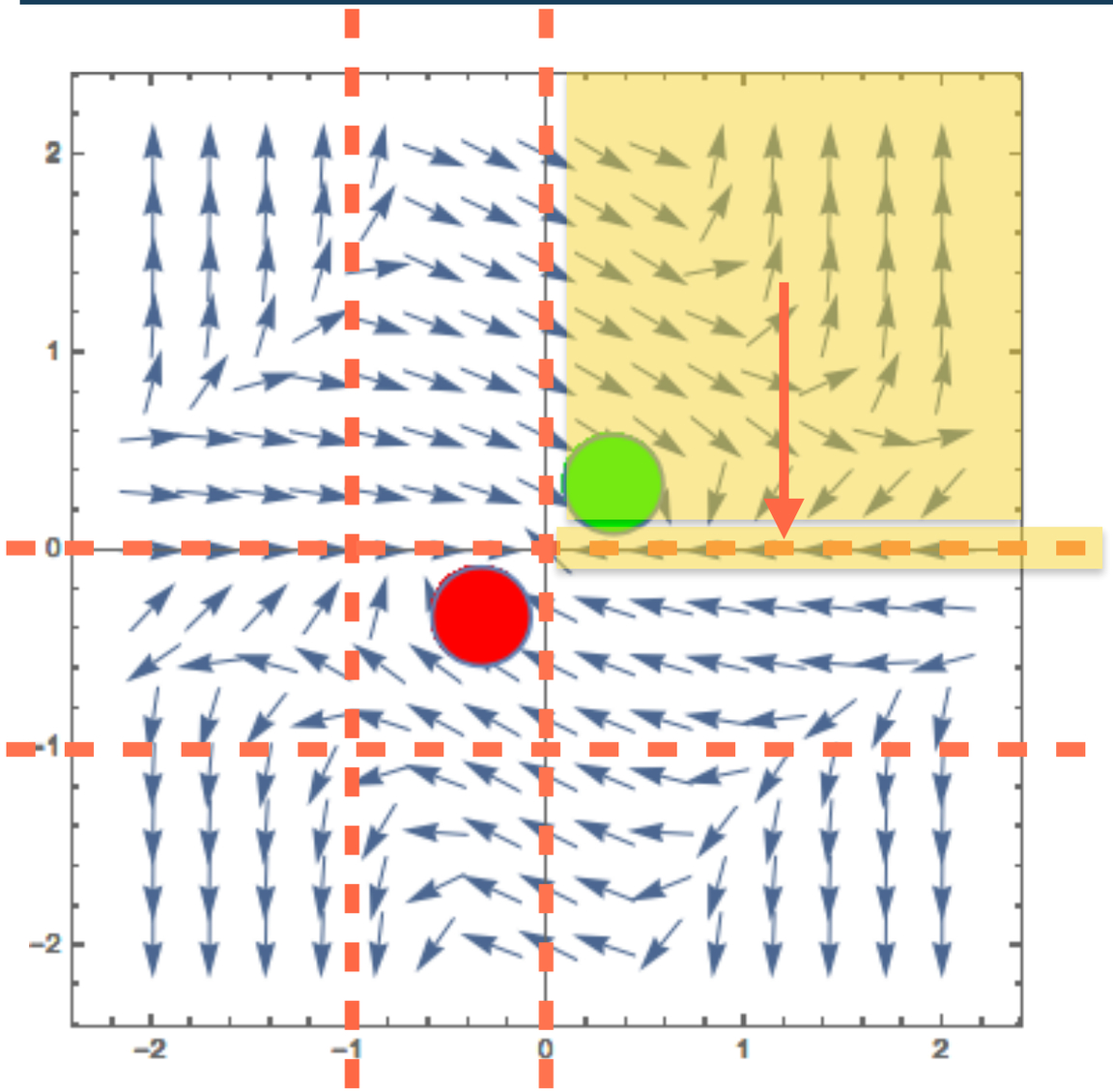


Check of existence: now is defined only among pairs of state (i.e., does s1 can reach s2?)

$$0 < x \wedge 0 < y \wedge 0 < x + 1 \wedge 0 < y + 1$$

$$0 < x \wedge 0 = y \wedge 0 < x + 1 \wedge 0 < y + 1$$

Main challenges



Check of existence: now is defined only among pairs of state (i.e., does s1 can reach s2?)

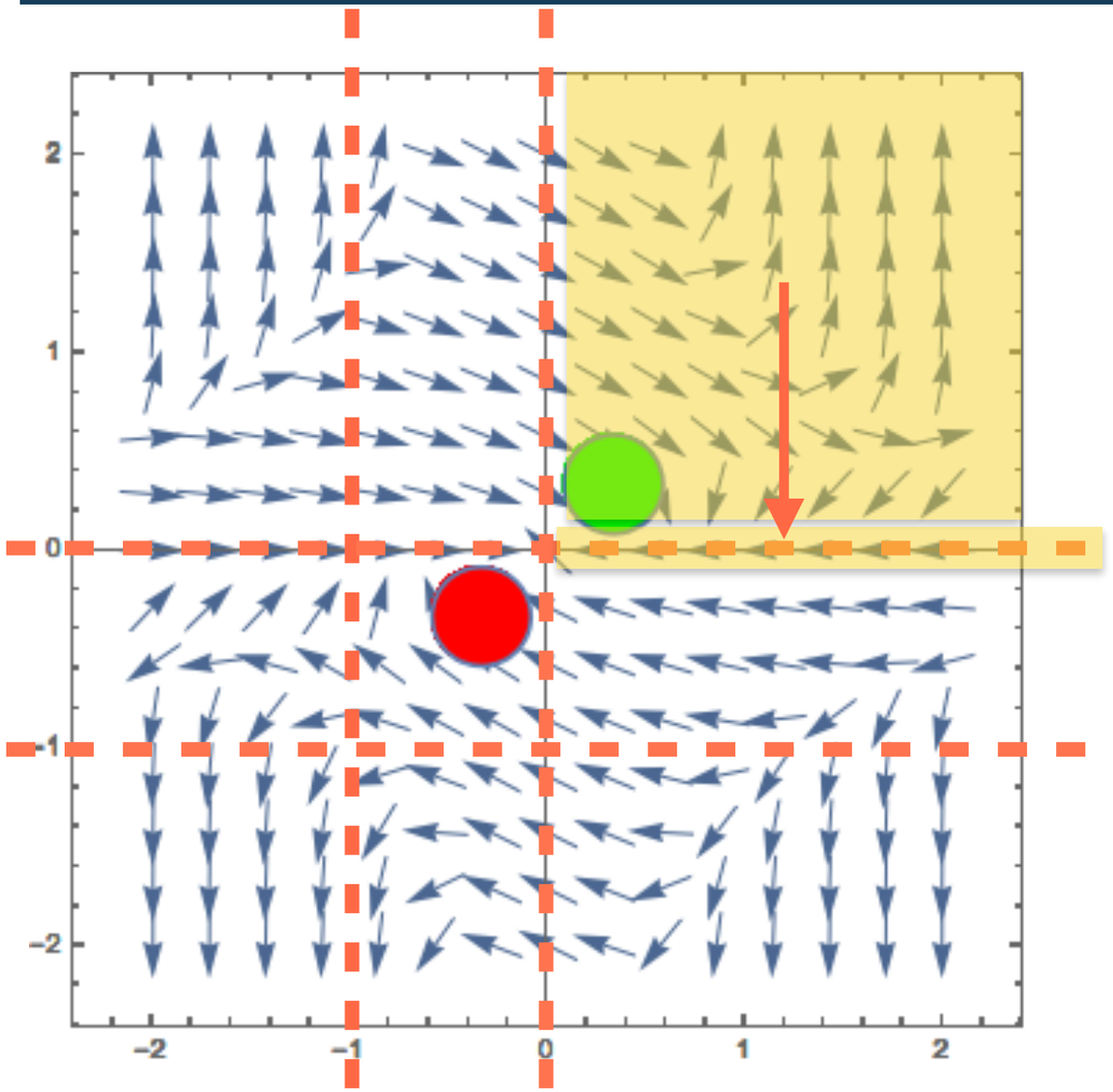
$$0 < x \wedge 0 < y \wedge 0 < x + 1 \wedge 0 < y + 1$$

$$s_1 \rightarrow [\dot{\vec{X}} = \vec{f}(\vec{X}) \;\&\; (s_1 \vee s_2)]\; s1$$

$$0 < x \wedge 0 = y \wedge 0 < x + 1 \wedge 0 < y + 1$$

Check of existence:
S1 is a differential invariant when the domain is restricted to s1 or s2

Main challenges



Check of existence: now is defined only among pairs of state (i.e., does s1 can reach s2?)

$$0 < x \wedge 0 < y \wedge 0 < x + 1 \wedge 0 < y + 1$$

$$s_1 \rightarrow [\dot{\vec{X}} = \vec{f}(\vec{X}) \ \& \ (s_1 \vee s_2)] \ s1$$

$$0 < x \wedge 0 = y \wedge 0 < x + 1 \wedge 0 < y + 1$$

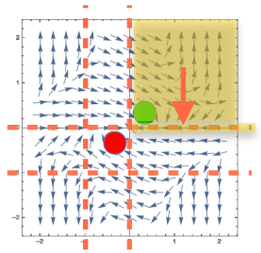
Check of existence:
S1 is a differential invariant when the domain is restricted to s1 or s2

Can we apply symbolic techniques to compute the semialgebraic decomposition?

In this talk

- Symbolic algebraic decomposition:
  - exponential encoding in the number of polynomials
- Linear-size algebraic decomposition
  - encoding linear in the number of polynomials
- Experimental evaluation

Exponential (symbolic) algebraic decomposition



Expressing the decomposition as a transition system

$\dot{X} = f(X)$ dynamical system

$I(X)$ Initial states

$\psi(X)$ Safe states

$A = \{a_1, a_2, \ldots, a_j\}$ Polynomials

We want to express the semialgebraic decomposition as a transition system

$$S_A := \langle V_A, Init(V_A), Trans(V_A, V_A') \rangle$$

$$P_A := \{a \bowtie 0 \mid a \in A \wedge \ \bowtie \in \{<, >, =\}\}$$

$$V_A := \{v_p \mid p \in P\}$$

$$I_A := \exists X.(I(X) \wedge \bigwedge_{p \in P} (v_p \leftrightarrow p(X)))$$

Transition relation of the decomposition (1/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg(s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1))$$

Transition relation of the decomposition (1/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg(s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1))$$

Encode all the possible transitions

Transition relation of the decomposition (1/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg(s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1))$$

Encode all the possible transitions

S1 to s2 if S1 is not invariant

Transition relation of the decomposition (1/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

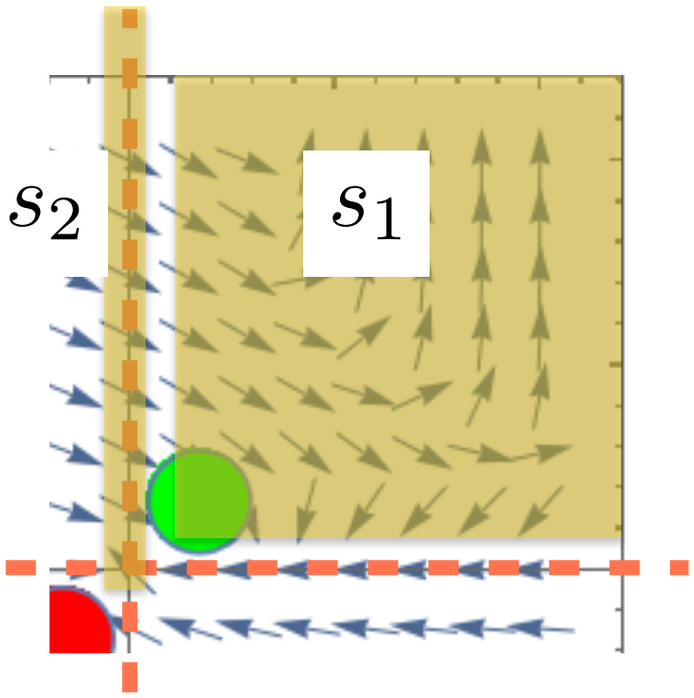
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg(s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1))$$

Encode all the possible transitions

S1 to s2 if S1 is not invariant

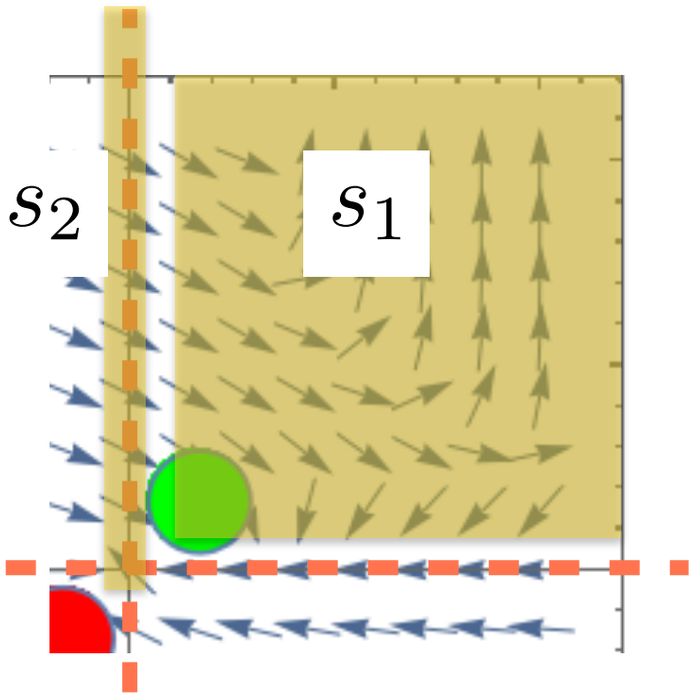
What is $s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$ **exactly?**

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

Checking differential invariants - the hard truth

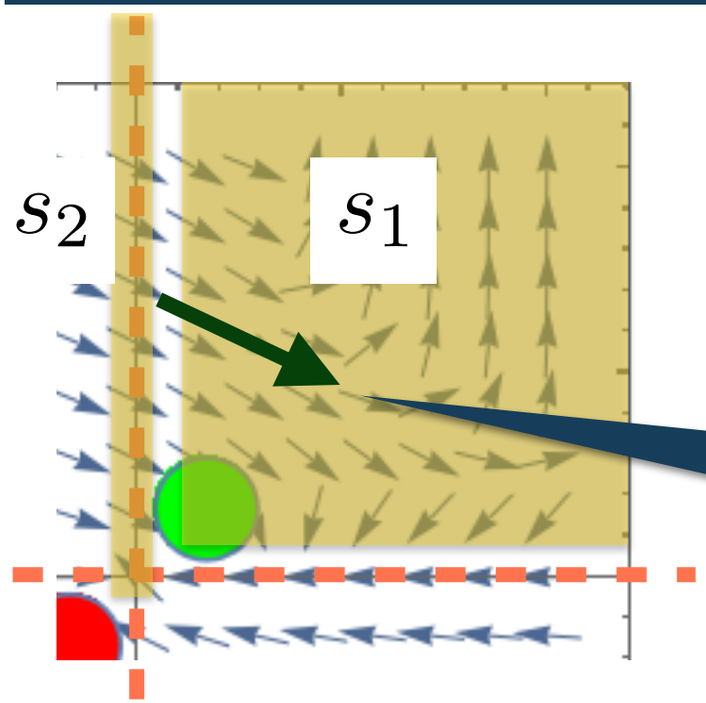


$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

Liu et al, EMSOFT11

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

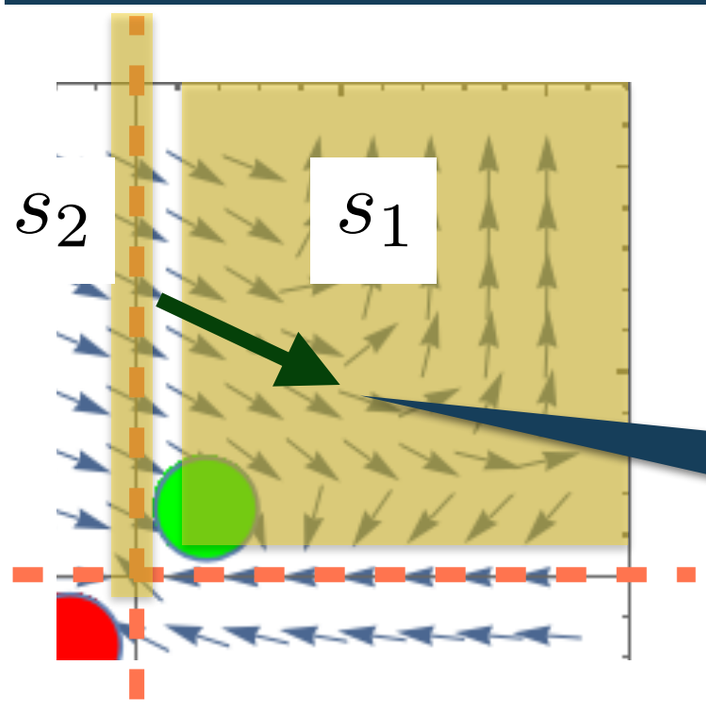
"LZZ" procedure

Liu et al, EMSOFT11

Some intuition: checks what happens on the border using the Lie derivative

$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} af \qquad L_f^1(x) > 0$$

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X)\ \&\ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

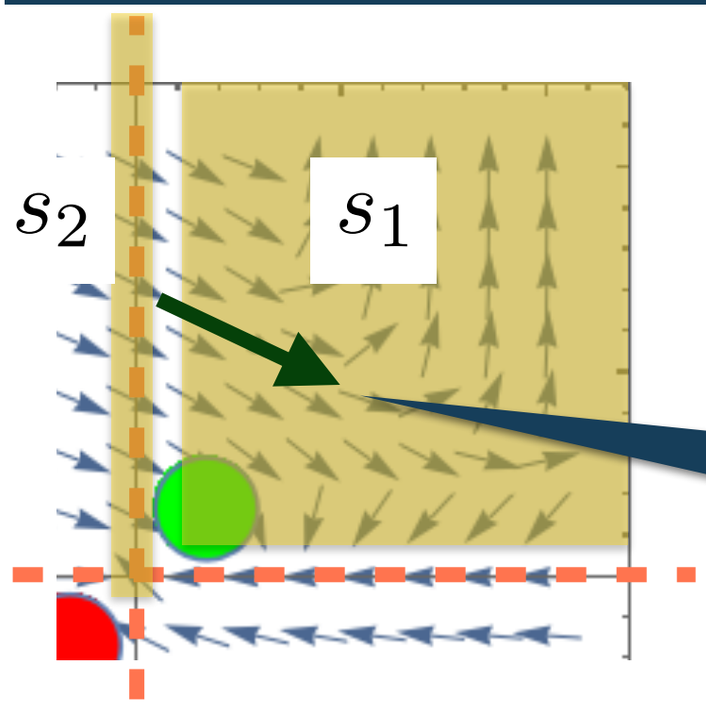
Liu et al, EMSOFT11

Some intuition: checks what happens on the border using the Lie derivative

$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} af \qquad L_f^1(x) > 0$$

What if a Lie derivative is 0 (somewhere)?

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

Liu et al, EMSOFT11

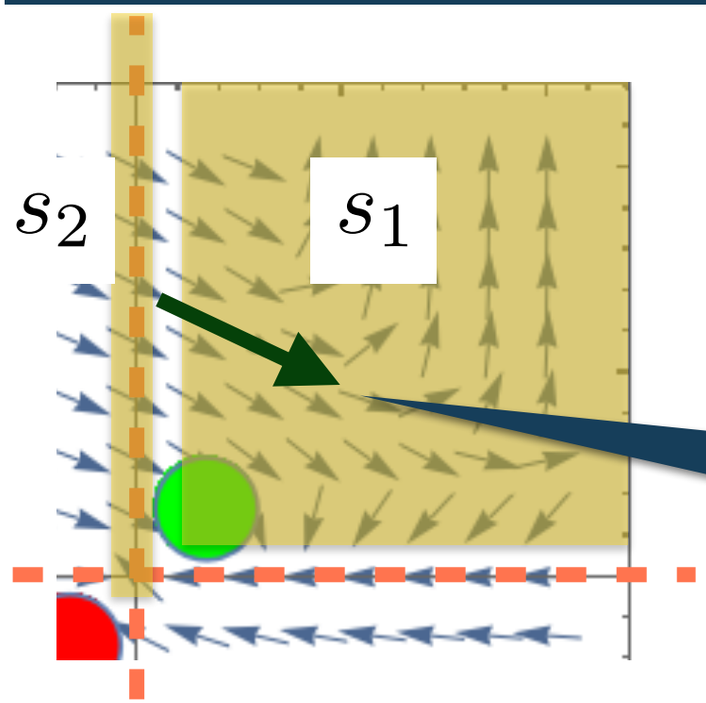
Some intuition: checks what happens on the border using the Lie derivative

$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} af \qquad L_f^1(x) > 0$$

What if a Lie derivative is 0 (somewhere)?

What if we have a semi-algebraic set? (boolean combination of predicates, issues with boundaries!)

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

Liu et al, EMSOFT11

Some intuition: checks what happens on the border using the Lie derivative

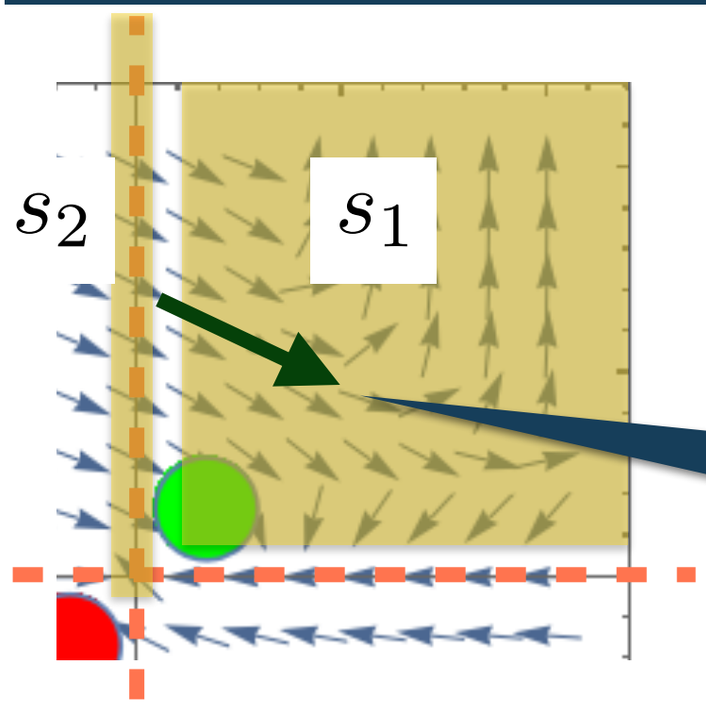
$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} a f \qquad L_f^1(x) > 0$$

What if a Lie derivative is 0 (somewhere)?

What if we have a semi-algebraic set? (boolean combination of predicates, issues with boundaries!)

$$LZZ_{s_1,f,s_1 \vee s_2}(\overline{X}) := \forall \overline{X}.((s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge In_{f,s_1 \vee s_2}(\overline{X})) \rightarrow In_{f,s_1}(\overline{X})) \wedge$$
$$((\neg s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge IvIn_{f,s_1 \vee s_2}(\overline{X})) \rightarrow \neg IvIn_{f,s_1}(\overline{X})))$$

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

Liu et al, EMSOFT11

Some intuition: checks what happens on the border using the Lie derivative

$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} af \qquad L_f^1(x) > 0$$

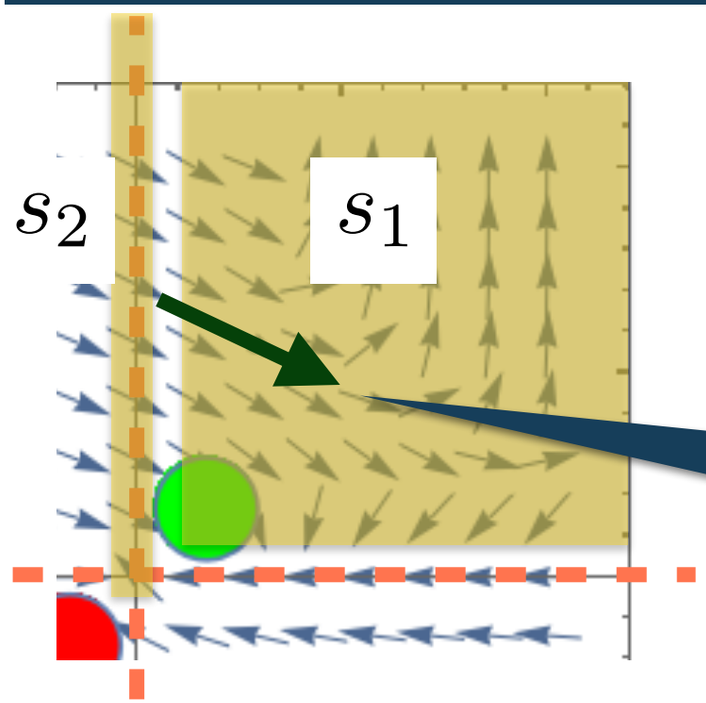
What if a Lie derivative is 0 (somewhere)?

What if we have a semi-algebraic set? (boolean combination of predicates, issues with boundaries!)

Conditions on the lie derivatives

$$LZZ_{s_1,f,s_1 \vee s_2}(\overline{X}) := \forall \overline{X}.((s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge In_{f,s_1 \vee s_2}(\overline{X})) \rightarrow In_{f,s_1}(\overline{X})) \wedge$$
$$((\neg s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge IvIn_{f,s_1 \vee s_2}(\overline{X})) \rightarrow \neg IvIn_{f,s_1}(\overline{X})))$$

Checking differential invariants - the hard truth



$$s_1 \rightarrow [\dot{X} = f(X) \ \& \ (s_1 \vee s_2)]s_1$$

"LZZ" procedure

Liu et al, EMSOFT11

Some intuition: checks what happens on the border using the Lie derivative

$$L_f^0 a = a \qquad L_f^1 a = \frac{\partial}{\partial \vec{X}} L_f^{i-1} af \qquad L_f^1(x) > 0$$

What if a Lie derivative is 0 (somewhere)?

What if we have a semi-algebraic set? (boolean combination of predicates, issues with boundaries!)

Conditions on the lie derivatives

$$LZZ_{s_1,f,s_1 \vee s_2}(\overline{X}) := \forall \overline{X}.((s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge In_{f,s_1 \vee s_2}(\overline{X})) \rightarrow In_{f,s_1}(\overline{X})) \wedge$$
$$((\neg s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge IvIn_{f,s_1 \vee s_2}(\overline{X})) \rightarrow \neg IvIn_{f,s_1}(\overline{X})))$$

Complete and sound (can be expressed as a semialgebraic set, i.e. nonlinear real arithmetic)

Transition relation of the decomposition (2/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \exists \overline{X}.\neg LZZ_{s_1,f,s_1 \vee s_2}(\overline{X})))$$

Transition relation of the decomposition (2/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \exists \overline{X}.\neg LZZ_{s_1,f,s_1 \vee s_2}(\overline{X})))$$

Encode that s1 moves to s2 (works because LZZ is complete)

Transition relation of the decomposition (2/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \exists \overline{X}. \neg LZZ_{s_1, f, s_1 \vee s_2}(\overline{X})))$$

Encode that s1 moves to s2 (works because LZZ is complete)

Issue: we still encode an exponential number of transitions

Transition relation of the decomposition (2/2)

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge$$

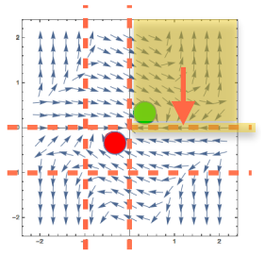
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \exists \overline{X}. \neg LZZ_{s_1, f, s_1 \vee s_2}(\overline{X})))$$

Encode that s1 moves to s2 (works because LZZ is complete)

Issue: we still encode an exponential number of transitions

The encoding is "trivial" but unfeasible

Linear algebraic decomposition



Restating our goal

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge \bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \exists \overline{X}. \neg LZZ_{s_1, f, s_1 \vee s_2}(\overline{X})))$$

- simplify LZZ
- encode the LZZ conditions "by predicate" instead of "by abstract state"

$$T_A := \exists X, X'.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge \mathbf{T}')$$

Linear in the number of polynomials

Simplifying LZZ

$$LZZ_{s_1,f,s_1 \vee s_2}(\overline{X}) := \forall \overline{X}.(\quad (s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge In_{f,s_1 \vee s_2}(\overline{X})) \rightarrow In_{f,s_1}(\overline{X})) \wedge$$
$$((\neg s_1(\overline{X}) \wedge ((s_1(\overline{X}) \vee s2(\overline{X})) \wedge IvIn_{f,s_1 \vee s_2}(\overline{X})) \rightarrow \neg IvIn_{f,s_1}(\overline{X})))$$

$$:= \ldots$$

By boolean simplifications and distributivity of In and IvIn operators

$$:= \forall \overline{X}.((\neg s_1(\overline{X}) \vee \neg In_{f,s_2}(\overline{X}) \vee In_{f,s_1}(\overline{X})) \wedge$$
$$(s_1(\overline{X}) \vee \neg s_2(\overline{X}) \vee \neg IvIn_{f,s_1}(\overline{X})))$$

"simpler formula"

$$\neg LZZ_{s1,f,s1 \vee s2} := (s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X})) \vee$$
$$(\neg s_1(\overline{X}) \wedge s_2(\overline{X}) \wedge IvIn_{f,s_1}(\overline{X})))$$

And in the end we want the negation of LZZ

Splitting the LZZ condition

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge \bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg LZZ_{s_1,f,s_1 \vee s_2}))$$

Again… some boring rewriting…

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge ($$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X})) \vee$$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg s_1(\overline{X}) \wedge s_2(\overline{X}) \wedge IvIn_{f,s_1}(\overline{X}))))$$

Splitting the LZZ condition

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge \bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg LZZ_{s_1,f,s_1 \vee s_2}))$$

Again… some boring rewriting…

What we gain: split the "forward" and "backward" checks of LZZ

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge ($$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X})) \vee$$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg s_1(\overline{X}) \wedge s_2(\overline{X}) \wedge IvIn_{f,s_1}(\overline{X}))))$$

Splitting the LZZ condition

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge \bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg LZZ_{s_1,f,s_1 \vee s_2}))$$

Again… some boring rewriting…

What we gain: split the "forward" and "backward" checks of LZZ

$$T_A := \exists X, X', \overline{X}.(\bigwedge_{p \in P} (v_p \leftrightarrow p(X)) \wedge \bigwedge_{p \in P} (v'_p \leftrightarrow p(X')) \wedge ($$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X})) \vee$$
$$\bigvee_{(s_1,s_2) \in 2^{Pa} \times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge \neg s_1(\overline{X}) \wedge s_2(\overline{X}) \wedge IvIn_{f,s_1}(\overline{X}))))$$

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow a(\overline{X}) \bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X') \bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow a(\overline{X}) \bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X') \bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

In distributes over a conjunction of predicates

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X') \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X}) \bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow a(\overline{X}) \bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X') \bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X) \bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

In distri
conjunctio

In distributes over a conjunction of predicates

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

In distri
conjunctio

In distributes over a conjunction of predicates

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

The condition on the LZZ check must hold every time the predicate hold

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

In distri
conjunctio

In distributes over a conjunction of predicates

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

The condition on the LZZ check must hold every time the predicate hold

Similarly for the In conditions

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

def. of abstract state

In distri
conjunctio

In distributes over a conjunction of predicates

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \rightarrow In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \rightarrow (\neg In_{f,a\bowtie 0}(\overline{X})$$

The condition on the LZZ check must hold every time the predicate hold

Similarly for the In conditions

Tricky for disjunction (it holds because we are considering all the predicates)

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

We have a similar encoding for the other "big disjunct"

def. of abstract state

In distri conjunctio


A PROOFS

$$InsExpl_f(X,X',\overline{X}) = \bigvee_{s_1,s_2\in 2^A} \Big(\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \Big(\bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X}) \Big) \Big) \quad (12)$$

$$InsSymb_f(X,X',\overline{X}) = \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big) \wedge \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big) \wedge \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big) \quad (13)$$

LEMMA 5.1. *The formulas $InsExpl_f(X,X',\overline{X})$ and $InsSymb_f(X,X',\overline{X})$ are equivalent.*

PROOF. We prove that the formulas $InsExpl_f(X,X',\overline{X})$ and $InsSymb_f(X,X',\overline{X})$ are equivalent.

$\Rightarrow$) We prove that $\models InsExpl_f(X,X',\overline{X}) \implies InsSymb_f(X,X',\overline{X})$

We show that a model μ of $InsExpl_f(X,X',\overline{X})$ (i.e., $\mu(X,X',\overline{X}) \models InsExpl_f(X,X',\overline{X})$) is also a model for $InsSymb_f(X,X',\overline{X})$ (i.e., $\mu(X,X',\overline{X}) \models InsSymb_f(X,X',\overline{X})$).

Since μ is a model for $InsExpl_f(X,X',\overline{X})$, then there's a disjunct in $InsExpl_f(X,X',\overline{X})$ such that $\mu \models InsExpl_f(X,X',\overline{X})$, there are two $s_1, s_2 \in 3^A$ such that:

$$\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \Big(\bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X}) \Big)$$

We have that $\mu \models InsSymb_f(X,X',\overline{X})$, since:

(1) $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big)$.
Consider a predicate $a \in A, \bowtie\in\{>,<,=\}$:
- When $\mu \models a(X)\bowtie 0$ then $a\bowtie 0 \in s_1$ (this is because $\mu \models a(X)\bowtie 0$ if and only of $a\bowtie 0 \in s_1$).
  Then we have both that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$ and $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0$, and so $\mu \models a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0$.
- When $\mu \models \neg a(X)\bowtie 0$, then we trivially have that $\mu \models a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0$.

(2) $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big)$
We can prove this case in a similar way to the above one.

(3) $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$
Since $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$ and $\mu \models \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X})$, then there exists $a\bowtie 0 \in s_1$ such that $\mu \models a\bowtie 0$ and $\mu \models \neg In_{f,a\bowtie 0}(\overline{X})$.
Thus, it's also the case that $\mu \models a\bowtie 0 \wedge \neg In_{f,a\bowtie 0}(\overline{X})$, implying $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$

$\Leftarrow$) We prove that $\models InsSymb_f(X,X',\overline{X}) \implies InsExpl_f(X,X',\overline{X})$.
- We first show that $\mu \models s_1(X) \wedge s_2(X')$, for some $s_1, s_2 \in 3^A$.
  – μ is a complete assignment to the variables $X, X', \overline{X}$ and for all $a \in A$ it is the case that $\mu \models a\bowtie 0$ exactly for one $\bowtie\in\{<,>,=\}$. Thus, we have that $s_1 = \{a\bowtie 0 \mid \mu \models a(X)\bowtie 0\}$ and that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$.
  – Similarly, we have that $s_2 = \{a\bowtie 0 \mid \mu \models a(X')\bowtie 0\}$ and $\mu \models \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0$.
- By hypothesis we have that $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big)$. Since $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$, then it follows that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0$.
- We similarly show that $\mu \models \bigwedge_{a\bowtie 0\in s_1} In_{f,a\bowtie 0}(\overline{X})$ (from $\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big)$ and $\mu \models \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0$.
- We have that $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$.
  So, there exists at least a predicate $a\bowtie 0$, $a \in A$ and $\bowtie\in\{>,<,=\}$, such that $\mu \models a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X}))$.
  We show that $a\bowtie 0 \in s_1$. Assume by absurd that $a\bowtie 0 \notin s_1$, meaning $\mu \not\models a(X)\bowtie 0$. Then, it means that there must exists a predicate $a\bowtie' 0 \in s_1$ such that $\bowtie\neq\bowtie'$ and that $\mu \models a(X)\bowtie' 0$, because s is a complete assignment and for each $a \in A$ exactly one among $a(X) > 0$, $a(X) = 0$, and $a(X) < 0$ holds. Clearly, this contradicts $\mu \models a(X)\bowtie 0$ (because $\mu \models a(X)\bowtie' 0 \implies \mu \not\models a(X)\bowtie 0$.

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \to a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \to In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \to (\neg In_{f,a\bowtie 0}(\overline{X})$$

The condition hold every t

Similarly for t

Tricky for disjunction (it holds because we are considering all the predicates)

Express each condition predicate-by-predicate

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (s_1(X) \wedge s_2(X') \wedge s_1(\overline{X}) \wedge In_{f,s_2}(\overline{X}) \wedge \neg In_{f,s_1}(\overline{X}))$$

$$= \bigvee_{(s_1,s_2)\in 2^{Pa}\times 2^{Pa}} (\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\triangleright}$$

We have a similar encoding for the other "big disjunct"

def. of abstract state

In distri
conjunctio


A PROOFS

$$InsExpl_f(X,X',\overline{X}) = \bigvee_{s_1,s_2\in 2^A} \Big(\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \Big(\bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X}) \Big) \Big) \quad (12)$$

$$InsSymb_f(X,X',\overline{X}) = \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big) \wedge \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big) \wedge \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big) \quad (13)$$

LEMMA 5.1. *The formulas $InsExpl_f(X,X',\overline{X})$ and $InsSymb_f(X,X',\overline{X})$ are equivalent.*

PROOF. We prove that the formulas $InsExpl_f(X,X',\overline{X})$ and $InsSymb_f(X,X',\overline{X})$ are equivalent.

$\Rightarrow$) We prove that $\models InsExpl_f(X,X',\overline{X}) \implies InsSymb_f(X,X',\overline{X})$

We show that a model μ of $InsExpl_f(X,X',\overline{X})$ (i.e., $\mu(X,X',\overline{X}) \models InsExpl_f(X,X',\overline{X})$) is also a model for $InsSymb_f(X,X',\overline{X})$ (i.e., $\mu(X,X',\overline{X}) \models InsSymb_f(X,X',\overline{X})$).

Since μ is a model for $InsExpl_f(X,X',\overline{X})$, then there's a disjunct in $InsExpl_f(X,X',\overline{X})$ such that $\mu \models InsExpl_f(X,X',\overline{X})$, there are two $s_1, s_2 \in 3^A$ such that:

$$\bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0 \wedge \bigwedge_{a\bowtie 0\in s_2} In_{f,a\bowtie 0}(\overline{X}) \wedge \Big(\bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X}) \Big)$$

We have that $\mu \models InsSymb_f(X,X',\overline{X})$, since:

(1) $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big)$.
Consider a predicate $a \in A, \bowtie\in\{>,<,=\}$:
- When $\mu \models a(X)\bowtie 0$ then $a\bowtie 0 \in s_1$ (this is because $\mu \models a(X)\bowtie 0$ if and only of $a\bowtie 0 \in s_1$).
  Then we have both that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$ and $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0$, and so $\mu \models a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0$.
- When $\mu \models \neg a(X)\bowtie 0$, then we trivially have that $\mu \models a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0$.

(2) $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big)$
We can prove this case in a similar way to the above one.

(3) $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$
Since $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$ and $\mu \models \bigvee_{a\bowtie 0\in s_1} \neg In_{f,a\bowtie 0}(\overline{X})$, then there exists $a\bowtie 0 \in s_1$ such that $\mu \models a\bowtie 0$ and $\mu \models \neg In_{f,a\bowtie 0}(\overline{X})$.
Thus, it's also the case that $\mu \models a\bowtie 0 \wedge \neg In_{f,a\bowtie 0}(\overline{X})$, implying $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$

$\Leftarrow$) We prove that $\models InsSymb_f(X,X',\overline{X}) \implies InsExpl_f(X,X',\overline{X})$.
- We first show that $\mu \models s_1(X) \wedge s_2(X')$, for some $s_1, s_2 \in 3^A$.
  – μ is a complete assignment to the variables $X, X', \overline{X}$ and for all $a \in A$ it is the case that $\mu \models a\bowtie 0$ exactly for one $\bowtie\in\{<,>,=\}$. Thus, we have that $s_1 = \{a\bowtie 0 \mid \mu \models a(X)\bowtie 0\}$ and that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$.
  – Similarly, we have that $s_2 = \{a\bowtie 0 \mid \mu \models a(X')\bowtie 0\}$ and $\mu \models \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0$.
- By hypothesis we have that $\mu \models \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \implies a(\overline{X})\bowtie 0 \Big)$. Since $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(X)\bowtie 0$, then it follows that $\mu \models \bigwedge_{a\bowtie 0\in s_1} a(\overline{X})\bowtie 0$.
- We similarly show that $\mu \models \bigwedge_{a\bowtie 0\in s_1} In_{f,a\bowtie 0}(\overline{X})$ (from $\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X')\bowtie 0 \implies In_{f,a\bowtie 0}(\overline{X}) \Big)$ and $\mu \models \bigwedge_{a\bowtie 0\in s_2} a(X')\bowtie 0$.
- We have that $\mu \models \bigvee_{a\in A,\bowtie\in\{>,<,=\}} \Big(a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X})) \Big)$.
  So, there exists at least a predicate $a\bowtie 0$, $a \in A$ and $\bowtie\in\{>,<,=\}$, such that $\mu \models a(X)\bowtie 0 \wedge (\neg In_{f,a\bowtie 0}(\overline{X}))$.
  We show that $a\bowtie 0 \in s_1$. Assume by absurd that $a\bowtie 0 \notin s_1$, meaning $\mu \not\models a(X)\bowtie 0$. Then, it means that there must exists a predicate $a\bowtie' 0 \in s_1$ such that $\bowtie\neq\bowtie'$ and that $\mu \models a(X)\bowtie' 0$, because μ is a complete assignment and for each $a \in A$ exactly one among $a(X) > 0$, $a(X) = 0$, and $a(X) < 0$ holds. Clearly, this contradicts $\mu \models a(X)\bowtie 0$ (because $\mu \models a(X)\bowtie' 0 \implies \mu \not\models a(X)\bowtie 0$.

$$= \bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \to a(\overline{X})\bowtie 0 \wedge$$

$$\bigwedge_{a\in A,\bowtie\in\{>,<,=\}} a(X')\bowtie 0 \to In_{f,a\bowtie 0}(\overline{X}) \wedge$$

$$\bigvee_{a\in A,\bowtie\in\{>,<,=\}} a(X)\bowtie 0 \to (\neg In_{f,a\bowtie 0}(\overline{X})$$

The condition
hold every t

Similarly for t

ause we
tes)

We get a linear encoding of the decomposition

Evaluation

Settings

- Non-linear safety verification benchmarks from Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016
  - Fix polynomial for the abstraction (factors and derivatives)

- Evaluate:
  - "Reach": Explicit reachability analysis Sogokon, Ghorbal, Jackson, Platzer, VMCAI 2016
    - Using different solvers: Mathematica, z3
  - "DWCL": find invariant predicates and call reach as subroutine
  - "ic3": linear encoding + ic3 model checking algorithm Cimatti et al TACAS 2014 Cimatti et al TACAS 2017

Conclusions

Conclusions

- First step to apply symbolic techniques to polynomial dynamics
- Mixed experimental results
  - Positive: safe vs. naive reachability computation
  - Negative: unsafe, vs. DWCL

Future Works

- Near future:
  - Extend the experiments
  - Understand bottlenecks in the verification algorithm (e.g., solver…)
  - Use more efficient formulation of LZZ (recent tech report from Sogokon and Ghorbail)
- Try to prove simple hybrid systems