Constraint-Based Static Analysis of Programs

Zohar Manna
Stanford University

Joint work with

Aaron Bradley, Michael Colón, Sriram Sankaranarayanan, Henny Sipma

Motivation

Objective: To extract information about the program behavior from the program text

- invariants
- ranking functions (termination)
- temporal properties
- . . .

Trivial Example

$$
\begin{array}{l}
\textbf{integer } i, j \textbf{ where } i = 2 \wedge j = 0 \\
\ell_0 : \ \textbf{while } (\ldots) \textbf{ do} \\
\quad \left[\begin{array}{l}
\textbf{if } (\ldots) \textbf{ then} \\
\qquad i := i + 4 \\
\textbf{else} \\
\qquad (i, j) := (i + 2, j + 1)
\end{array} \right]
\end{array}
$$

$i \geq 2$, $j \geq 0$, and $i - 2j \geq 2$ are invariants.

Objective: To obtain such invariants automatically

Buffer Overflow Analysis

```
1: int *a = malloc( sizeof(int) * n);
2: int i,j,k;
3: for(i=0; i<n; ++i)
4:       for(j=0;2*j<=i;++j)
5:           if (a[i] <= a[2*j+1])
6:                .....
7:       ...
```

$0 \leq i < n$?

$0 \leq 2j + 1 < n$?

Check bounds for each array access.

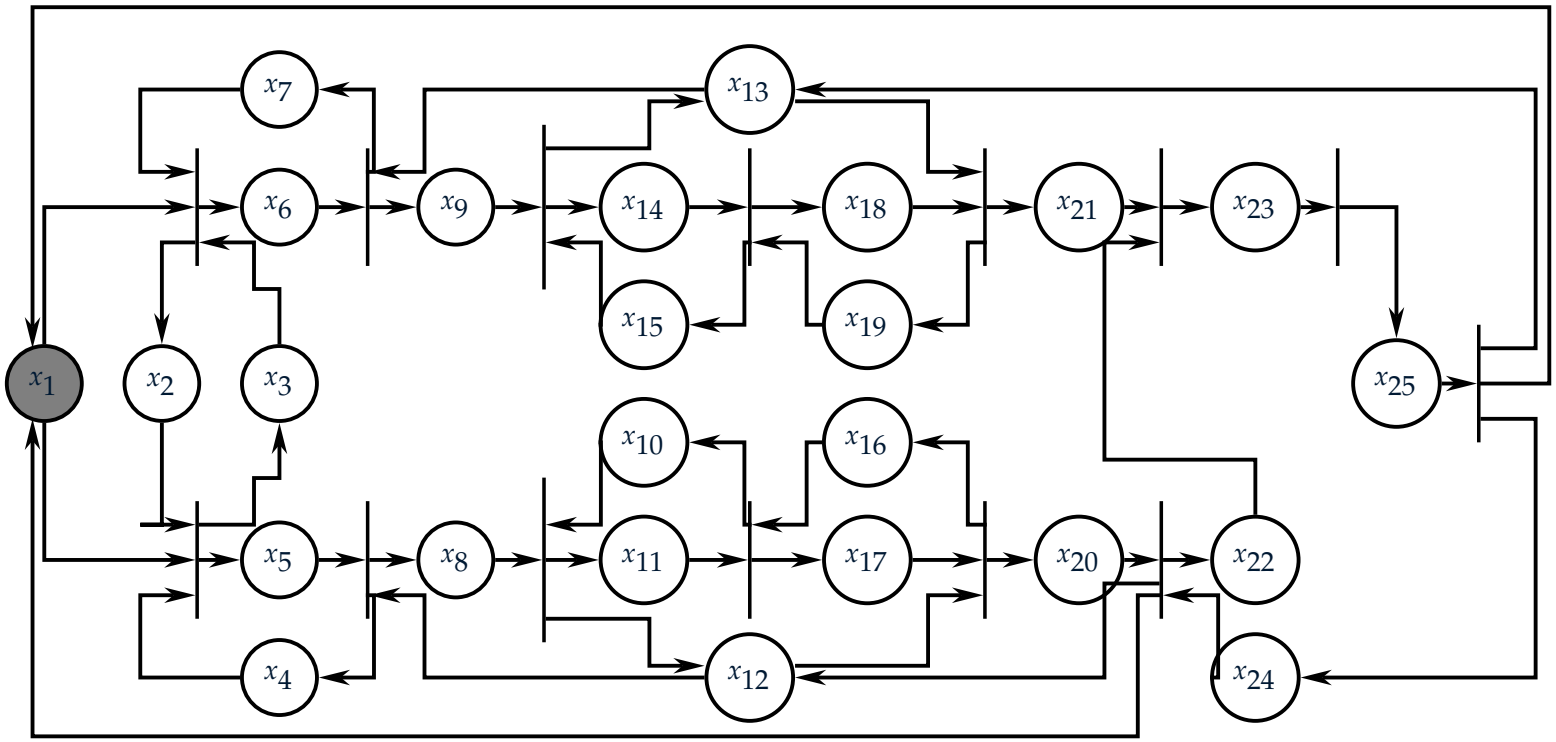
Division by Zero

```
1: double a,b,c
2: ....
3: while  ( b > 0 || c >= 0 ) {
4:          a = a + b/(c+b-1);
5: ....
6: }
```

$c + b - 1 > 0$

Prove every divisor non-zero.

Deadlock Freedom



Is this Petri net deadlock free?

[Zhou et al. : 1992]

Termination

local i, j, k: **integer**

while ($i \leq 100 \ \wedge \ j \leq k$) **do**

$(i, j, k) \ := \ (j, i+1, k-1)$

od

Termination is proved by the ranking function

$$-i - j + k$$

which decrements by 2 each iteration.

We need the supporting invariant

$$-i - j + k \geq 0$$

to establish termination.

Preliminaries: Transition Systems

$$\textbf{integer } i, j \textbf{ where } i = 2 \wedge j = 0$$
$$l_0 : \textbf{ while true do}$$
$$\left[\begin{array}{c} i := i + 4 \\ \textbf{or} \\ (i, j) := (i + 2, j + 1) \end{array}\right]$$

Transition system:

$$\Big\langle \underbrace{L : \{\ell_0\}}_{\textit{locations}},\ \underbrace{V : \{i, j\}}_{\textit{variables}},\ \underbrace{\mathcal{T} : \{\tau_1, \tau_2\}}_{\textit{transitions}},\ \underbrace{\Theta : (i = 2 \wedge j = 0)}_{\textit{initial condition}},\ \underbrace{L_0 : l_0}_{\textit{initial location}} \Big\rangle$$

with

$$\tau_1 = \langle l_0,\ l_0,\ \rho_{\tau_1} : (i' = i + 4 \wedge j' = j) \rangle$$
$$\tau_2 = \Big\langle l_0,\ l_0,\ \rho_{\tau_2} : \underbrace{(i' = i + 2 \wedge j' = j + 1)}_{\textit{transition relation}} \Big\rangle$$

Transition System: Execution

$$\langle L,\ V,\ \mathcal{T},\ \Theta,\ L_0 \rangle$$

Computation: Infinite sequence of states $\langle l_i, \boldsymbol{x_i} \rangle$

$$\langle \ell_0, \boldsymbol{x_0} \rangle \xrightarrow{\tau_1} \langle \ell_1, \boldsymbol{x_1} \rangle \xrightarrow{\tau_2} \langle \ell_2, \boldsymbol{x_2} \rangle \rightarrow \cdots$$

such that

- Initial Condition satisfied

$$\ell_0 = L_0 \ \wedge \ \Theta(\boldsymbol{x_0})$$

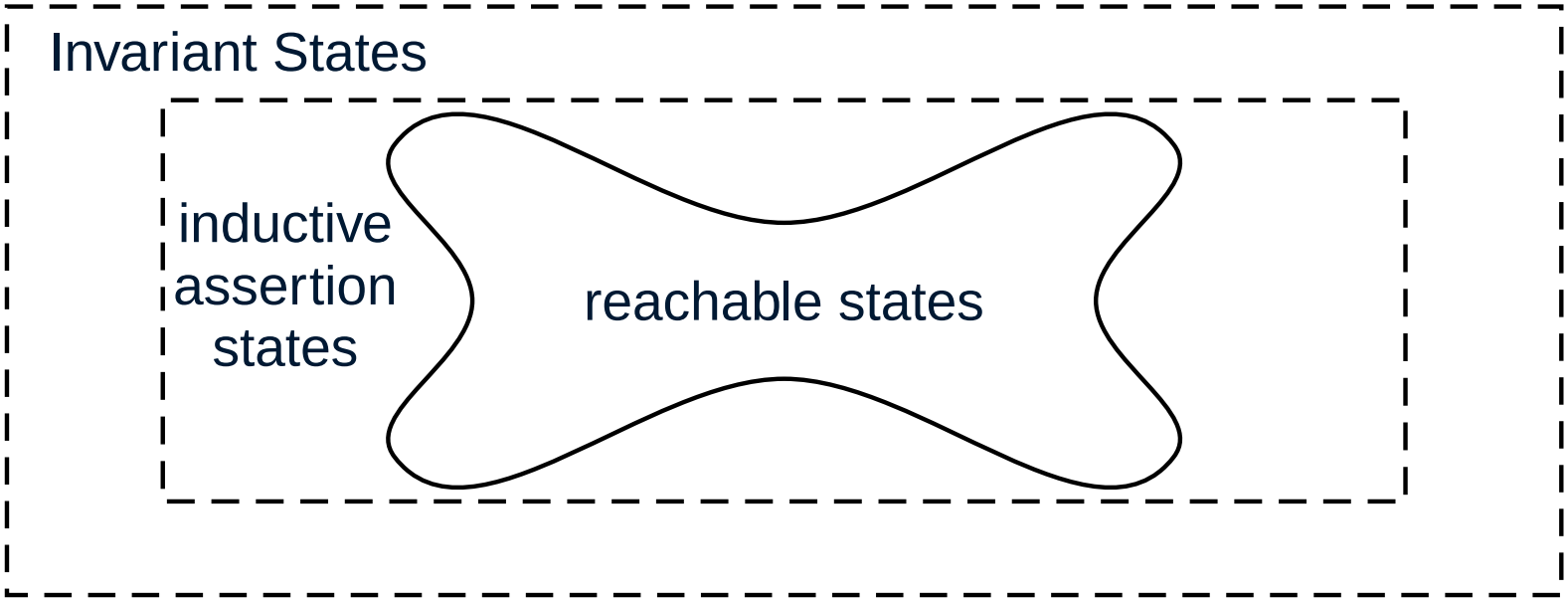
- Consecutive states $\langle \ell_i, \boldsymbol{x_i} \rangle \rightarrow \langle \ell_{i+1}, \boldsymbol{x_{i+1}} \rangle$ satisfy some transition

$$\tau_k : \ \langle \ell_i,\ \ell_{i+1},\ \rho_{\tau_k}(\boldsymbol{x_i}, \boldsymbol{x_{i+1}}) \rangle$$

Invariants

Assertion ψ is an *invariant of P* iff

it is true at all the *reachable states* of P.



Example:

reachable states : $\langle \ell_0, 2 \rangle, \langle \ell_0, 4 \rangle, \langle \ell_0, 8 \rangle, \langle \ell_0, 16 \rangle, \ldots$

invariant : $at\_\ell_0 \rightarrow x$ is even

Inductive Assertions

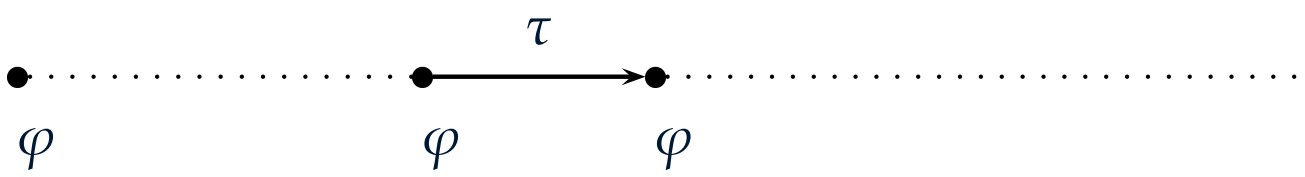
An assertion φ is inductive assertion iff

Initiation The assertion is true initially,

$$\Theta \models \varphi$$

Consecution For every transition τ, if φ is true before τ is taken, then it is true after taking τ,

$$\varphi \wedge \rho_\tau \models \varphi'$$



Invariant vs. Inductive Assertion

Inductive Assertion $\Rightarrow$ Invariant

Invariant $\not\Rightarrow$ Inductive Assertion

To prove invariant φ, find inductive assertion ψ, satisfying initiation + consecution, such that

$$\psi \rightarrow \varphi$$

To find invariants,
we really search for inductive assertions.

Example

integer i, j **where** $i = 2 \wedge j = 0$

while true do

$$l_0 : \left[\begin{array}{c} i := i + 4 \\ \textbf{or} \\ (i, j) := (i + 2, j + 1) \end{array} \right]$$

Is $\boxed{\varphi : \; i - 2j \geq 2}$ an invariant?

Example: Continued

Show

Initiation: $\underbrace{(i = 2 \ \wedge \ j = 0)}_{\Theta} \models \underbrace{i - 2j \geq 2}_{\varphi}$

Consecution: Two transitions τ_1 and τ_2.

$$\underbrace{i - 2j \geq 2}_{\varphi} \ \wedge \ \underbrace{(i' = i + 4 \ \wedge \ j' = j)}_{\rho_{\tau_1}} \models \underbrace{i' - 2j' \geq 2}_{\varphi'}$$

$$\underbrace{i - 2j \geq 2}_{\varphi} \ \wedge \ \underbrace{(i' = i + 2 \ \wedge \ j' = j + 1)}_{\rho_{\tau_2}} \models \underbrace{i' - 2j' \geq 2}_{\varphi'}$$

$\boxed{\varphi : \ i - 2j \geq 2}$ is inductive assertion $\Rightarrow$ invariant.

Preliminaries: Assertion Domains

Assertion Domain:

A class of assertions, containing: the initial assertion,
the transition relations, and
the target invariants.

Common Examples:

1. Linear Equalities over Reals $2i + j - 3 = 0,$
2. Linear Inequalities over Reals (convex polyhedra) $2i + 3j - 2 \leq 0,$
3. Integer Arithmetic (Presburger Arithmetic) $(\exists\, k)[2j = i \ \wedge \ i = j + k],$
4. Multiplication over Reals $(\exists a, b)[ai^3 + bi = j \ \vee \ i = j^2].$

Linear Invariants

Invariants in the domain of Linear Equalities or Inequalities are called *Linear Invariants*.

All variables and constants are assumed to range over the reals.

Example:

$$\textbf{integer } i, j \textbf{ where } i = 2 \wedge j = 0$$
$$\ell_0 : \textbf{ while } (\ldots) \textbf{ do}$$
$$\left[\begin{array}{l} \textbf{if } (\ldots) \textbf{ then} \\ \quad i := i + 4 \\ \textbf{else} \\ \quad (i, j) := (i + 2, j + 1) \end{array}\right]$$

$\boxed{\varphi_1 : \; i \geq 2}$ and $\boxed{\varphi_2 : \; i - 2j \geq 2}$ are linear invariants.

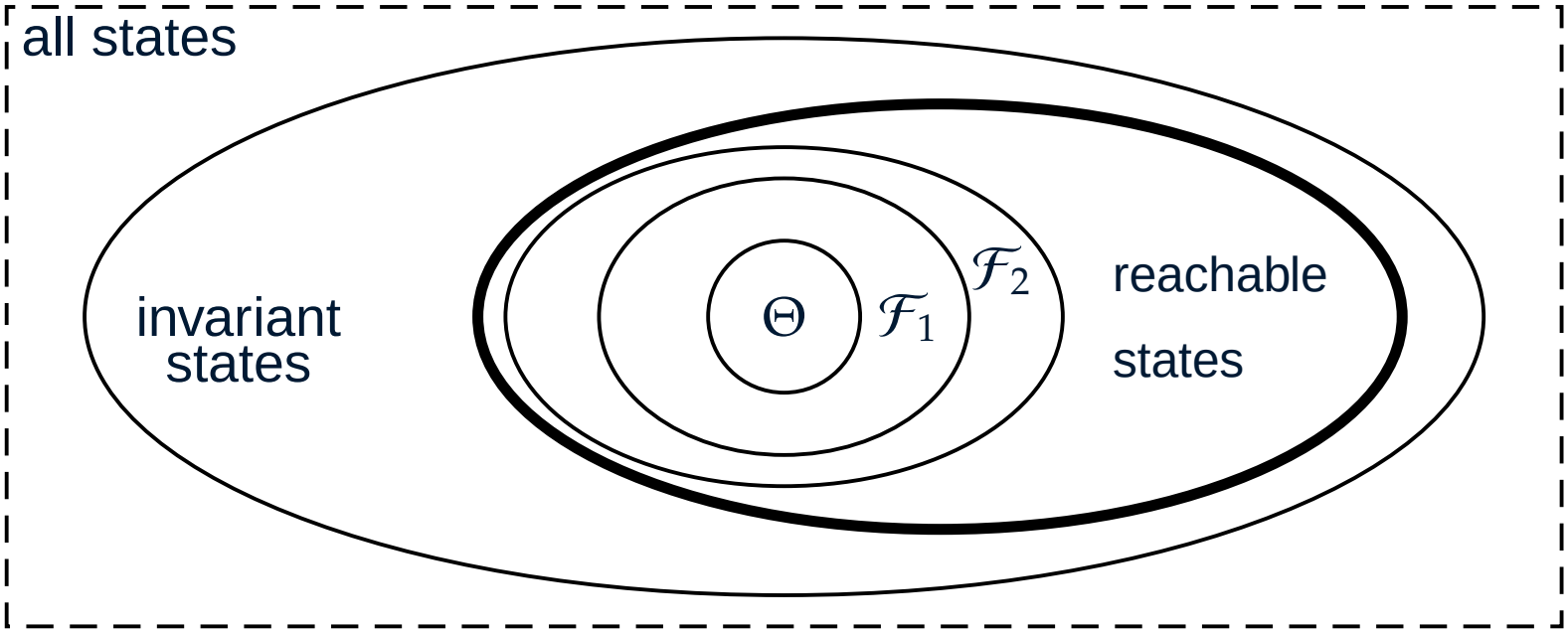
"i is even" is an invariant but not linear.

Linear Relations Analysis

Static Analysis: Traditional Approach

Goal: Given a program, find invariants

Symbolic forward simulation to obtain an overapproximation of the reachable state space (i.e. invariants)

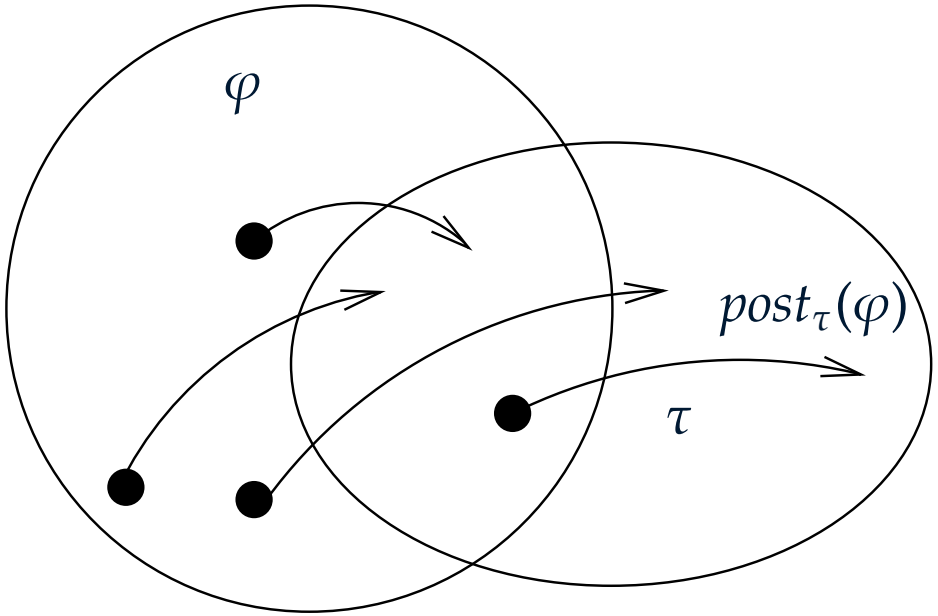


Forward Propagation

$$
\begin{array}{lcl}
\mathcal{F}_0 & : & \Theta \\
\mathcal{F}_1 & : & \mathcal{F}_0 \ \vee \ (\bigvee_{\tau \in \mathcal{T}} post_\tau(\mathcal{F}_0)) \\
\mathcal{F}_2 & : & \mathcal{F}_1 \ \vee \ (\bigvee_{\tau \in \mathcal{T}} post_\tau(\mathcal{F}_1)) \\
\vdots & &
\end{array}
$$

until $\mathcal{F}_{i+1} \rightarrow \mathcal{F}_i$ (use widening operator to force convergence)

where $post_\tau(\varphi) : \ \exists V_0 \ \cdot (\varphi(V_0) \wedge \rho_\tau(V_0, V))$



Problems

1. May not converge in finite time
   Example:

$$\textbf{integer } i \textbf{ where } i = 0$$
$$\textbf{while true do } i := i + 1$$

$$\begin{aligned} \mathcal{F}_0 &: \quad i = 0 \\ \mathcal{F}_1 &: \quad i = 0 \ \vee \ i = 1 \\ \mathcal{F}_2 &: \quad i = 0 \ \vee \ i = 1 \ \vee i = 2 \\ &\vdots \end{aligned}$$

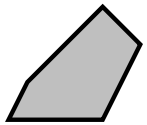
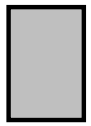
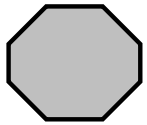
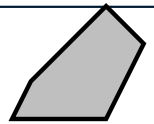
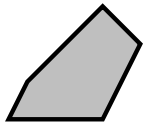
   We never reach: $i \geq 0$

2. May not be able to detect convergence

$$\mathcal{F}_{n+1} \rightarrow \mathcal{F}_n$$

Common Solution

Abstract Interpretation [Cousot&Cousot,77]: perform the symbolic simulation in an abstract domain:

Reference	Shape	Invariants (over reals)
Karr ’76 Müller-Olm, Seidl, ’04 Gulwani, Necula ’03		$a_1x_1 + \cdots + a_nx_n = b$ (linear equalities)
Cousot, Halbwachs ’79		$a_1x_1 + \cdots + a_nx_n \leq b$ (linear inequalities)
Cousot, Cousot ’76		$\ell \leq x_i \leq u$ (intervals)
Mine ’01		$x_i - x_j \leq b$ (octagons)
Clarisó, Cortadella ’04		$\sum a_ix_i \leq \ell,\ a_i \in \{-1,0,1\}$ (octahedra)
Sankaranarayanan, Sipma, Manna ’04		$\underline{a_1}x_1 + \ldots + \underline{a_n}x_n \leq b$ $\underline{a_i}$ fixed

Example: Forward Propagation

$$\textbf{integer } i, j \textbf{ where } i = 2 \wedge j = 0$$

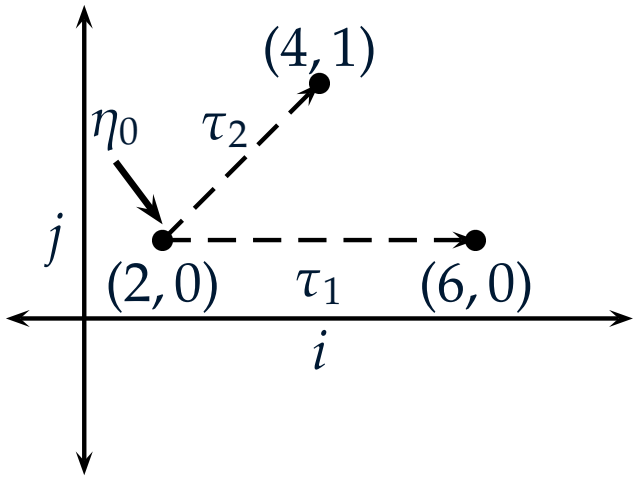
$$l_0 : \textbf{ while true do}$$

$$\left[\begin{array}{c} i := i + 4 \\ \textbf{or} \\ (i, j) := (i + 2, j + 1) \end{array}\right]$$

Domain: Linear Inequalities over Reals

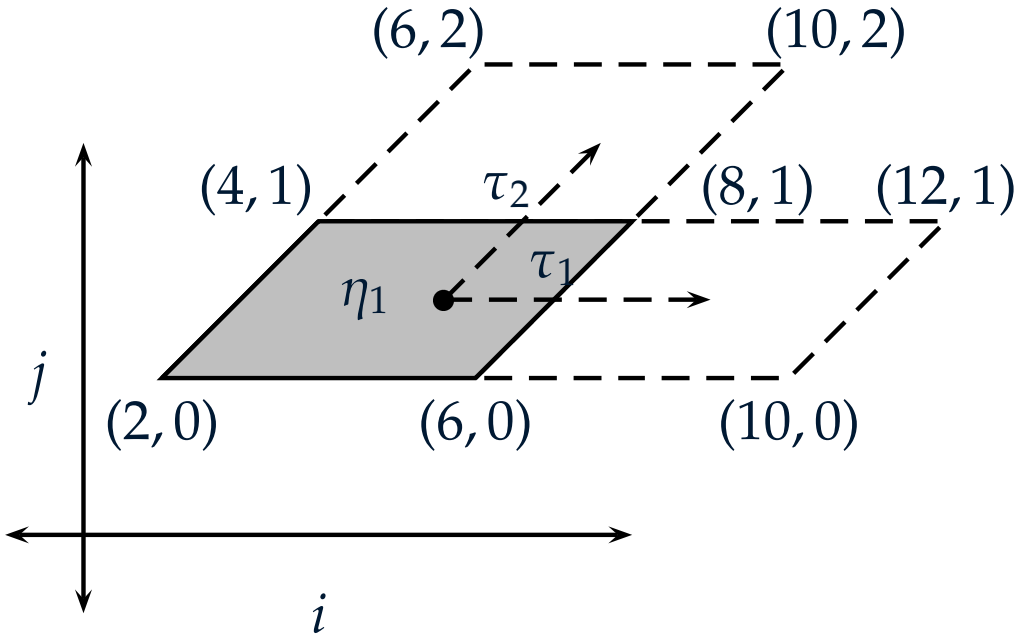
Step 1: Iteration

$$
\begin{aligned}
\eta_0 : &\quad (j = 0) \quad \wedge \quad (i = 2) \\
post(\eta_0, \tau_1) : &\quad (j = 0) \quad \wedge \quad (i = 6) \\
post(\eta_0, \tau_2) : &\quad (j = 1) \quad \wedge \quad (i = 4) \\
\eta_1 : &\quad (0 \leq j \leq 1) \quad \wedge \quad (2 \leq i - 2j \leq 6)
\end{aligned}
$$



Step 2: Iteration

$$
\begin{aligned}
\eta_1 : &\quad (0 \le j \le 1) \quad \wedge \quad (2 \le i - 2j \le 6) \\
post(\eta_1, \tau_1) : &\quad (0 \le j \le 1) \quad \wedge \quad (6 \le i - 2j \le 10) \\
post(\eta_1, \tau_2) : &\quad (1 \le j \le 2) \quad \wedge \quad (2 \le i - 2j \le 6) \\
\eta_2 : &\quad (0 \le j \le 2) \quad \wedge \quad (2 \le i - 2j \le 10)
\end{aligned}
$$

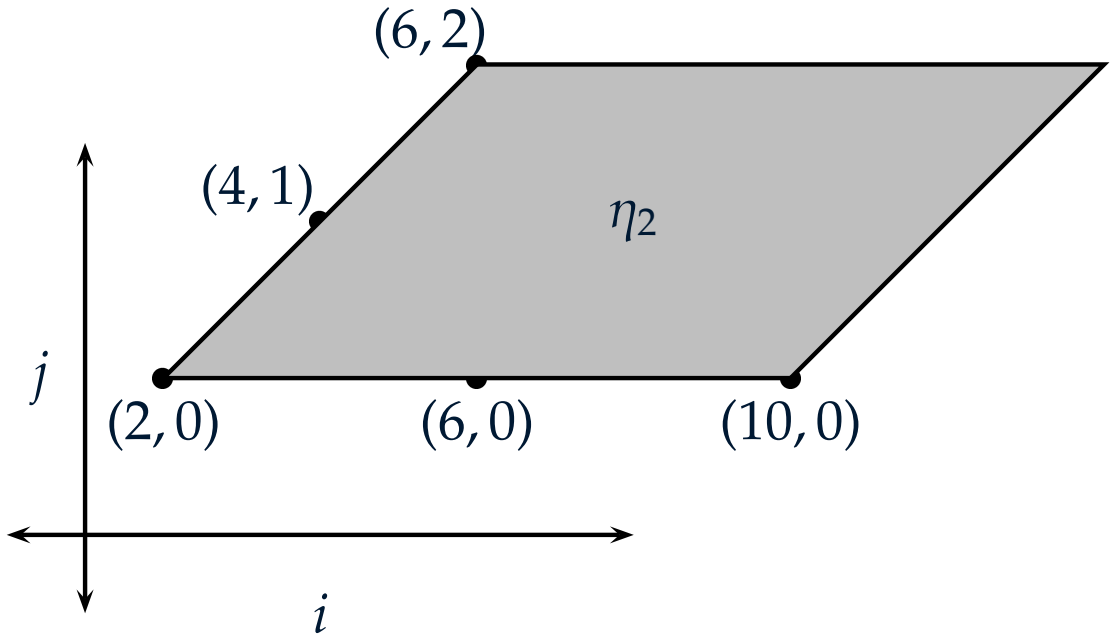


Step 3: Widening Iteration

$$\eta_1: \quad \underline{(0 \le j \le 1)} \quad \wedge \quad \underline{(2 \le i - 2j \le 6)}$$

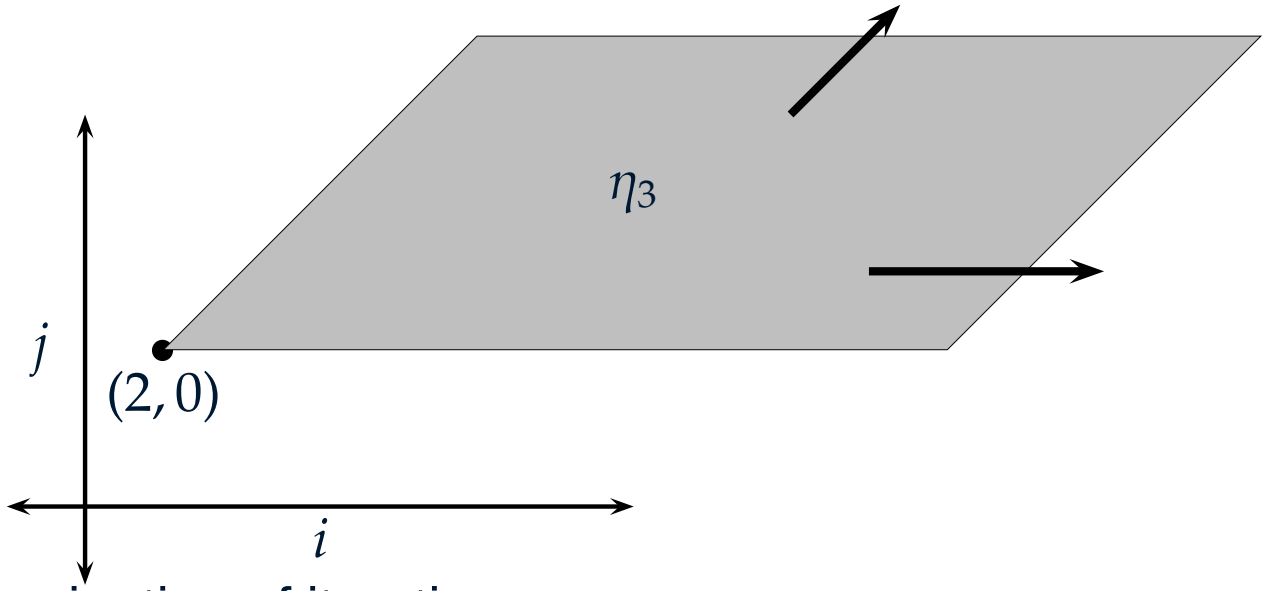
$$\eta_2: \quad \underline{(0 \le j \le 2)} \quad \wedge \quad \underline{(2 \le i - 2j \le 10)}$$

$$\eta_3(\text{widening}): \quad (0 \le j) \quad \wedge \quad (2 \le i - 2j)$$



Iteration: Step 4

$$
\begin{aligned}
\eta_3 : &\quad (0 \le j) \quad \wedge \quad (2 \le i - 2j) \\
post(\eta_3, \tau_1) : &\quad (0 \le j) \quad \wedge \quad (2 \le i - 2j) \\
post(\eta_3, \tau_2) : &\quad (0 \le j) \quad \wedge \quad (2 \le i - 2j) \\
\eta_4 : &\quad \underline{(0 \le j)} \quad \wedge \quad \underline{(2 \le i - 2j)}
\end{aligned}
$$



Note: Termination of iteration, $\eta_4 = \eta_3$.

The final invariants are $\boxed{0 \le j} \wedge \boxed{2 \le i - 2j} \Rightarrow \boxed{i \ge 2}$.

Constraint-based Analysis

Constraint-based Analysis: Overview

1. Fix the domain and template of the desired invariant

   Examples:
   - linear invariant over reals
   - polynomial invariant over reals

2. Provide the conditions for the invariant to hold

3. Encode the conditions on the invariant as a system of constraints

4. Solve the constraints

5. Every solution is an invariant of the desired domain and template

Computing Linear Invariants

1. Fix domain and template

- **Domain:** Linear inequalities over reals
- **Template** (target invariant) **:**

$$c_1 x_1 + c_2 x_2 + \ldots + c_n x_n + d \leq 0$$

where

$\{x_1, \ldots, x_n\}$ are the program variables

and

$\{c_1, \ldots, c_n, d\}$ are unknown coefficients

2. Invariant Conditions

The property

$$\psi : \; c_1 x_1 + c_2 x_2 + \ldots + c_n x_n + d \leq 0$$

is an invariant of transition system

$$\Phi : \langle L, \; V : \{x_1, \ldots, x_n\}, \; \Theta, \; \mathcal{T} : \{\tau_1, \ldots, \tau_k\}, \; L_0 \rangle$$

if

$$\begin{array}{rcll} \Theta & \models & \psi & \text{(initiation)} \\ \psi \wedge \rho_{\tau_1} & \models & \psi' & \left.\begin{array}{l} \\ \\ \\ \end{array}\right\} \text{(consecution)} \\ \vdots & & & \\ \psi \wedge \rho_{\tau_k} & \models & \psi' & \end{array}$$

that is, if

- it is implied by the initial condition, and
- it is preserved by all transitions of the system

Invariant Conditions: Example

$$
\begin{array}{l}
\textbf{integer } i, j \textbf{ where } i = 2 \wedge j = 0 \\
l_0 : \ \textbf{while true do} \\
\qquad \left[\begin{array}{c} i := i + 4 \\ \textbf{or} \\ (i, j) := (i + 2, j + 1) \end{array} \right]
\end{array}
$$

Target invariant: $\boxed{\psi : \ c_1 i + c_2 j + d \leq 0}$

Conditions:

$$
\begin{array}{lllll}
 & \underbrace{i = 2 \wedge j = 0}_{\Theta} & \models & \underbrace{c_1 i + c_2 j + d \leq 0}_{\psi} & \\
c_1 i + c_2 j + d \leq 0 & \wedge \quad i' = i + 4 \wedge j' = j & \models & c_1 i' + c_2 j' + d \leq 0 & \\
\underbrace{c_1 i + c_2 j + d \leq 0}_{\psi} & \wedge \quad \underbrace{i' = i + 2 \wedge j' = j + 1}_{\rho_{\tau_1},\, \rho_{\tau_2}} & \models & \underbrace{c_1 i' + c_2 j' + d \leq 0}_{\psi'} &
\end{array}
$$

Farkas's Lemma

Let S be a system of linear inequalities over real-valued variables $x_1, \ldots, x_n$,

$$S: \begin{bmatrix} a_{11}x_1 & + & \cdots & + & a_{1n}x_n & + & b_1 & \leq 0 \\ \vdots & & & & \vdots & & \vdots & \\ a_{m1}x_1 & + & \cdots & + & a_{mn}x_n & + & b_m & \leq 0 \end{bmatrix}$$

and ψ a linear inequality,

$$\psi: \; c_1x_1 + \cdots + c_nx_n + d \leq 0$$

If S is satisfiable, $S \models \psi$ iff there exist *real multipliers* $\lambda_1, \ldots, \lambda_m \geq 0$ such that:

$$c_1 = \sum_{i=1}^{m} \lambda_i a_{i1} \quad \ldots \quad c_n = \sum_{i=1}^{m} \lambda_i a_{in} \quad d \leq (\sum_{i=1}^{m} \lambda_i b_i)$$

3. Encode the conditions: Initiation

Initiation:

$$\Theta \models c_1 x_1 + \cdots + c_n x_n + d \leq 0$$

is encoded by

$$\begin{array}{c|ccccccc} \lambda_1 & a_{11}x_1 & + & \cdots & + & a_{1n}x_n & + & b_1 \quad \leq 0 \\ \vdots & \vdots & & & & \vdots & & \vdots \\ \lambda_m & a_{m1}x_1 & + & \cdots & + & a_{mn}x_n & + & b_m \quad \leq 0 \\ \hline & c_1 x_1 & + & \cdots & + & c_n x_n & + & d \quad \leq 0 \end{array} \Bigg\} \Theta$$

which produces the constraints

$$S_0 : \exists(\lambda_1 \ldots \lambda_m \geq 0) \left(\begin{array}{ll} c_1 = \sum_{i=1}^{m} \lambda_i a_{i1} & \wedge \\ \cdots & \wedge \\ c_n = \sum_{i=1}^{m} \lambda_i a_{in} & \wedge \\ d \leq \sum_{i=1}^{m} \lambda_i b_i & \end{array} \right)$$

Example: Encoding Initiation

Target invariant $\boxed{\psi:\ c_1 i + c_2 j + d \le 0}$.

Initial Condition: $\underbrace{i = 2 \ \wedge \ j = 0}_{\Theta} \models \underbrace{c_1 i + c_2 j + d \le 0}_{\psi}$

λ_1	i			-2	$= 0$	Θ	i	$-$	2	≤ 0
λ_2		j			$= 0$		$-i$	$+$	2	≤ 0
	$c_1 i$	$+$	$c_2 j$	$+$	d	≤ 0	$\leftarrow \psi$			

$$\exists\, \lambda_1, \lambda_2 \ [\lambda_1 = c_1 \ \wedge \ \lambda_2 = c_2 \ \wedge \ d \le -2\lambda_1]$$

No requirement $\lambda_1, \lambda_2 \ge 0$!

The constraint after elimination of the λ's is

$$\boxed{S_0:\ 2c_1 + d \le 0}$$

Example: Encoding Consecution for τ_1

$$\underbrace{c_1 i + c_2 j + d \leq 0}_{\psi} \;\wedge\; \underbrace{i' = i + 4 \;\wedge\; j' = j}_{\rho_{\tau_1}} \;\models\; \underbrace{c_1 i' + c_2 j' + d \leq 0}_{\psi'}$$

μ_1	$c_1 i$	$+$	$c_2 j$					$+$	d	≤ 0	$\leftarrow$	ψ	
λ_1	i			$-$	i'			$+$	4	$= 0$			ρ_{τ_1}
λ_2			j			$-$	j'			$= 0$			
					$c_1 i'$	$+$	$c_2 j'$	$+$	d	≤ 0	$\leftarrow$	ψ'	

Constraints:

$$(\exists \mu_1 \geq 0)(\exists \lambda_1, \lambda_2)\left(\begin{array}{c} \mu_1 c_1 + \lambda_1 = 0 \;\wedge\; \mu_2 c_2 + \lambda_2 = 0 \wedge \\ -\lambda_1 = c_1 \;\wedge\; -\lambda_2 = c_2 \;\wedge\; d \leq \mu_1 d + 4\lambda_1 \end{array} \right)$$

Eliminating $\mu_1, \lambda_1, \lambda_2$:

$$\boxed{S_1 : \; (c_1 \leq 0) \;\vee\; (c_1 = 0 \;\wedge\; c_2 = 0)}$$

Example: Combined Constraint

The overall constraint is:

$$
\begin{array}{cl}
(2c_1 + d \leq 0) & \leftarrow \textit{Initiation} \\
\wedge & \\
\left[\begin{array}{l} (c_1 \leq 0) \vee \\ (c_1 = 0 \wedge c_2 = 0) \end{array}\right] & \leftarrow \tau_1 \textit{ consecution} \\
\wedge & \\
\left[\begin{array}{l} (2c_1 + c_2 \leq 0) \vee \\ (c_1 = 0 \wedge c_2 = 0) \end{array}\right] & \leftarrow \tau_2 \textit{ consecution}
\end{array}
$$

which simplifies to

$$\boxed{2c_1 + d \leq 0 \ \wedge \ c_1 \leq 0 \ \wedge \ 2c_1 + c_2 \leq 0}$$

4. Solve the constraints

Solve the constraint systems

$$S_0 \ \wedge \ S_1 \ \wedge \ \ldots \ \wedge \ S_k$$

for $\{c_1, \ldots, c_n, d\}$

Example: Solving the Constraints

The basic solutions of

$$2c_1 + d \leq 0 \ \wedge \ c_1 \leq 0 \ \wedge \ 2c_1 + c_2 \leq 0$$

are

c_1	c_2	d	$c_1 i + c_2 j + d \leq 0$
0	0	-1	$-1 \leq 0$
0	-1	0	$-j \leq 0$
-1	2	2	$-i + 2j + 2 \leq 0$

which corresponds to the inductive invariants

$\boxed{j \geq 0}$ and $\boxed{i - 2j \geq 2}$ $\Rightarrow$ $\boxed{i \geq 2}$

5. Solutions

For all solutions of $\{c_1, \ldots, c_n, d\}$,

$$c_1 x_1 + \ldots + c_n x_n + d \leq 0$$

is an invariant.

- **Good news**:
  The method is complete for linear systems (over reals).

  The solutions of $\{c_1, \ldots, c_n\}$ represent all inductive invariants that are linear inequalities of the given template.

Summary

1. Fix a *target invariant* with unknown coefficients,

$$c_1 i + c_2 j + d \leq 0$$

2. Encode the invariant conditions
   (initiation and consecution for each transition)
3. Compute constraints on the unknown coefficients,

$$2c_1 + d \leq 0 \ \wedge \ c_1 \leq 0 \ \wedge \ 2c_1 + c_2 \leq 0$$

4. Solve these constraints

$$\langle c_1, c_2, d \rangle = \langle 0, -1, 0 \rangle \qquad \langle c_1, c_2, d \rangle = \langle -1, 2, 2 \rangle$$

5. Generate the invariants

$$\begin{aligned} \langle 0, -1, 0 \rangle &\leftrightarrow 0i - 1j + 0 \leq 0 \\ \langle -1, 2, 2 \rangle &\leftrightarrow -1i + 2j + 2 \leq 0 \end{aligned}$$

Invariants: $\boxed{j \geq 0}$ and $\boxed{i - 2j \geq 2} \Rightarrow \boxed{i \geq 2}$

Pros and Cons

Advantages:

- No widening necessary
- All inductive invariants are generated (or obtained as consequences)
- System structure can be exploited to obtain linear constraints: Petri nets
- Properties other than invariants

Disadvantages:

- The constraint systems $S_1, \ldots, S_k$ are nonlinear and may be hard to solve. Tool: QEPCAD [Hong 93] (Cylindrical Algebraic Decomposition)
- But: $S_1, \ldots, S_k$ are parametric linear (cx okay, but not x^2)
  More efficient solution methods:
  factorization, polynomial root finding
  Tool: REDLOG [Weispfennig 92; Dolzmann, Sturm 97]

Computing Linear Ranking Functions

1. Fix domain and template

- **Domain:** Linear ranking functions over reals
- **Template:**

$$c_1 x_1 + c_2 x_2 + \ldots + c_n x_n + d$$

where

$\{x_1, \ldots, x_n\}$ are the program variables

and

$\{c_1, \ldots, c_n, d\}$ are unknown coefficients

2. Property Conditions

The function

$$\delta : \; c_1 x_1 + c_2 x_2 + \ldots + c_n x_n + d$$

is a ranking function of a loop

$$\Phi : \langle L,\; V : \{x_1, \ldots, x_n\},\; \Theta,\; \mathcal{T} : \{\tau_1, \ldots, \tau_k\},\; L_0 \rangle$$

if

$$\underbrace{\begin{array}{ll} \rho_{\tau_1} & \models \delta \geq 0 \\ \vdots & \\ \rho_{\tau_k} & \models \delta \geq 0 \end{array}}_{\text{bounded}} \qquad \underbrace{\begin{array}{ll} \rho_{\tau_1} & \models \delta - \delta' \geq \epsilon \\ \vdots & \\ \rho_{\tau_k} & \models \delta - \delta' \geq \epsilon \end{array}}_{\text{ranking}}$$

for some $\epsilon > 0$; that is, if

- it is bounded from below, and
- it is decreased by each transition

3. Encode the conditions

$$\delta : \; c_1 x_1 + \cdots + c_n x_n + d$$

Use Farkas's Lemma:

- Bounded:

$$B_i : \; \rho_{\tau_i} \models \delta \geq 0$$

- Ranking

$$R_i : \; \rho_{\tau_i} \models \delta - \delta' \geq \epsilon$$

  for some $\epsilon > 0$

4. Solve the constraints

Solve the constraint systems

$$B_1 \wedge \ldots \wedge B_k \ \wedge \ R_1 \wedge \ldots \wedge R_k$$

for $\{c_1, \ldots, c_n, d\}$

5. Solutions

The function

$$c_1 x_1 + c_2 x_2 + \ldots + c_n x_n + d$$

is a ranking function for all solutions of $\{c_1, \ldots, c_n, d\}$.

- **Good news**:
  The method is complete for linear systems over reals.
  The solutions represent all linear ranking functions of the given (uninitialized) loop.
- **Good news**:
  Constraints are all linear: can be solved efficiently
- **Bad news**:
  Most ranking functions require supporting invariants to prove boundedness (combine!)

$$I \wedge \rho_{\tau_i} \models \delta \geq 0$$

Computing Nonlinear Invariants

GCD-LCM Example

$\text{GCD}(x_1, x_2)$- Greatest Common Divisor of x_1 and x_2,

$$\text{GCD}(14, 21) = 7, \quad \text{GCD}(13, 21) = 1$$

$\text{LCM}(x_1, x_2)$- Least Common Multiple of x_1 and x_2,

$$\text{LCM}(14, 21) = 42, \quad \text{LCM}(13, 21) = 273$$

$$\text{GCD}(x_1, x_2) \cdot \text{LCM}(x_1, x_2) = x_1 \cdot x_2$$

GCD-LCM Example

integer $x_1, x_2, y_1, y_2, y_3, y_4$ **where**

$$(\; y_1 = x_1 \; \wedge \; y_2 = y_3 = x_2 \; \wedge \; y_4 = 0 \;)$$

ℓ_0 : **while** $(y_1 \neq y_2)$ **do** $\cdots\cdots$ $y_1 y_3 + y_2 y_4 = x_1 x_2$

$$\left[\begin{array}{l} \ell_1 : \textbf{ while } (y_1 > y_2) \textbf{ do} \\ \quad (y_1, y_4) := (y_1 - y_2, y_4 + y_3) \\ \ell_2 : \textbf{ while } (y_2 > y_1) \textbf{ do} \\ \quad (y_2, y_3) := (y_2 - y_1, y_3 + y_4) \end{array} \right]$$

$$\{y_1 = \mathrm{GCD}(x_1, x_2), \; y_3 + y_4 = \mathrm{LCM}(x_1, x_2)\}$$

Template

$$\underbrace{\underline{c_1}x_1x_2 + \underline{c_2}y_1x_2 + \underline{c_3}y_2x_1 + \underline{c_4}y_1y_3 + \underline{c_5}y_2y_4 + \underline{c_6}}_{p} = 0$$

Question: For what values of $\underline{c_1}, \ldots, \underline{c_6}$ is $p = 0$ invariant at ℓ_0?

Goal: Find values of $\underline{c_1}, \ldots, \underline{c_6}$ such that

- Initiation: $\Theta \models p = 0,$
- Consecution: $p = 0 \wedge \rho_{\tau_1} \models p' = 0,$
  $p = 0 \wedge \rho_{\tau_2} \models p' = 0.$

The Goal

To find $c_1, \ldots, c_6$ such that

$$p = 0 \, ,$$

construct $c_1, \ldots, c_6$ such that

initiation + consecution

are satisfied.

The Problem:
How do we encode

$$p_1 = 0 \ \wedge \ p_2 = 0 \ \wedge \ \cdots \ \wedge \ p_m = 0 \ \models \ p = 0 \qquad \cdots (A)$$

where p_i, p are polynomials?

We shall use Gröbner bases.

Linear Algebra

Linear Equalities (over reals):

$$e: \; 2x + 3y + \frac{4}{5}z - 4$$

Problem: When does

$$\underbrace{(e_1 = 0 \;\wedge\; e_2 = 0 \;\wedge\; \cdots \wedge\; e_m = 0)}_{\text{Linear Equalities}} \;\models\; \underbrace{e = 0}_{\text{Linear Equality}} \; ? \qquad \cdots (A)$$

Answer: $e = \underline{\lambda_1} e_1 + \underline{\lambda_2} e_2 + \cdots + \underline{\lambda_m} e_m \qquad \cdots (B)$

for some $\underline{\lambda_1}, \ldots, \underline{\lambda_m}$, real multipliers.

$(A) \;\Leftrightarrow\; (B)$

Algorithm: *Gaussian Elimination*

Linear Programming

Linear inequalities (over reals):

$$e: \; 2x + 3y + \frac{4}{5}z - 4$$

Problem: When does

$$\underbrace{(e_1 \leq 0 \;\wedge\; e_2 \leq 0 \;\wedge\; \cdots \wedge\; e_m \leq 0)}_{\text{Linear Inequalities}} \;\models\; \underbrace{e \leq 0}_{\text{Linear Inequality}} \; ? \qquad \cdots (A)$$

Answer (Farkas's Lemma) **:** $e = \lambda_1 e_1 + \lambda_2 e_2 + \cdots + \lambda_m e_m \quad \cdots (B)$
for some $\lambda_1, \ldots, \lambda_m \geq 0$, real multipliers.

$(A) \;\Leftrightarrow\; (B)$

Algorithm: *Cylindrical Algebraic Decomposition*

Algebraic Geometry

Polynomials (over reals):

$$3x^3y^3 + \frac{7}{9}x^2y^2z^2 + \frac{3}{4}y^2$$

Problem: When does

$$\underbrace{p_1 = 0 \ \wedge \ p_2 = 0 \ \wedge \ \cdots \ \wedge \ p_m = 0}_{\text{polynomial equalities}} \models \underbrace{p = 0}_{\text{polynomial equality}} \quad ? \quad \cdots (A)$$

Answer: $p = g_1p_1 + g_2p_2 + \cdots + g_mp_m \quad \cdots (B)$

for some $g_1, \ldots, g_m$, arbitrary polynomial multipliers.

$(B) \Rightarrow (A)$

Algorithm: *Gröbner bases and normal form reduction*.

Ideals

Ideal: The ideal generated by

$$P = \{p_1, \ldots, p_m\}$$

is the set of all polynomials of the form

$$\boxed{\textsc{Ideal}(P) = \{g_1 p_1 + \ldots + g_m p_m \mid g_1, \ldots, g_m \text{ polynomials}\}}$$

Example: Let $P = \{x^2 - y,\ y - z,\ x + z\}$.

$$\textsc{Ideal}(P) = \left\{ \begin{array}{l} g_1(x^2 - y) + g_2(y - z) + g_3(x + z) \mid \\ \quad g_1, g_2, g_3 \text{ are polynomials over } x, y, z \end{array} \right\}$$

$$-zx - z = \underbrace{1}_{g_1} \cdot (x^2 - y) + \underbrace{1}_{g_2} \cdot (y - z) + \underbrace{-x}_{g_3} (x + z)$$

Therefore, $-zx - z \in \textsc{Ideal}(P)$.

Ideal Membership

Goal: Given polynomials $P = \{p_1, \ldots, p_m\}$ and p,

Decide

$$(p_1 = 0 \ \wedge \ p_2 = 0 \ \wedge \ \cdots \ \wedge \ p_m = 0) \ \models \ (p = 0) \qquad \cdots \ (A)$$

Solution: Test if

$$p \in \text{IDEAL}(\underbrace{\{p_1, p_2, \ldots, p_m\}}_{P}) \qquad \cdots \ (B)$$

i.e, $p = g_1 p_1 + \cdots + g_m p_m$ for some polynomials $g_1, \ldots, g_m$.

- We cannot efficiently test (A).
- We know from Algebraic Geometry that $(B) \ \Rightarrow \ (A)$.

Question: How do we test (B) efficiently?

Testing Ideal Membership

Given any set of polynomials $P = \{p_1, \ldots, p_m\}$ and p,

How do we test if $p \in \text{IDEAL}(P)$? $\cdots (B)$

1. Compute Gröbner basis G of P (independent of p),
i.e, set of polynomials $G = \{p'_1, \ldots, p'_k\}$, such that

- $\text{IDEAL}(G) = \text{IDEAL}(P)$,
- G-rules $\xrightarrow{G}$ are confluent and terminating.

Use Büchberger's Algorithm + Refinements.

2. Apply the G-rules to p. It leads to
unique normal form $\text{NF}_G(p)$, $p \xrightarrow{G} \cdots \xrightarrow{G} \text{NF}_G(p)$.

Theorem: $p \in \text{IDEAL}(P) \cdots (B)$
iff
$\text{NF}_G(p) = 0. \quad \cdots (C)$

Testing Ideal Membership: Example

Let $P = \{p_1 : x^2 - y,\ p_2 : y + z,\ p_3 : x - z\}$.
Can we find out if

$$x^2 - z \in \textsc{Ideal}(P)$$

using $\xrightarrow{P}$? No!

Gröbner basis of P is

$$G = \{z^2 - z, y - z, x + z\}$$

Can we find out using $\xrightarrow{G}$? Yes!

Any sequence of G-reductions $\xrightarrow{G}$ from

$$p :\ x^2 - z$$

has normal form 0. Therefore

$$x^2 - z \in \textsc{Ideal}(P)$$

Template Constraints

Let $P = \{\underbrace{x^2 - y}_{p_1}, \underbrace{y + z}_{p_2}, \underbrace{x - z}_{p_3}\}$.

Problem: For what values of $c_1, c_2, \ldots, c_5$ does

$$p_1 = 0 \ \wedge \ p_2 = 0 \ \wedge \ p_3 = 0 \models \underbrace{c_1 x^2 + c_2 y^2 + c_3 z^2 + c_4 z + c_5}_{p} = 0 \ ?$$

Solution:

1. Compute the Gröbner basis of P,

$$G = \{z^2 - z, \ y - z, \ x + z\}$$

2. Compute normal form of p using G-rules,

$$\mathsf{NF}_G(p) = (c_1 + c_2 + c_3 + c_4)z + c_5$$

3. Set every coefficient to be zero,

$$(c_1 + c_2 + c_3 + c_4 = 0) \ \wedge \ (c_5 = 0)$$

Template Constraints (Cont)

Note: For solutions to $\underline{c_1}, \ldots, \underline{c_5}$ that satisfy

$$(\underline{c_1} + \underline{c_2} + \underline{c_3} + \underline{c_4} = 0) \ \wedge \ (\underline{c_5} = 0)$$

it follows that $\text{NF}_G(p) = 0$

therefore, $\underbrace{\underline{c_1}x^2 + \underline{c_2}y^2 + \underline{c_3}z^2 + \underline{c_4}z + \underline{c_5}}_{p} \in \text{IDEAL}(P) \qquad \cdots(B)$

therefore, $p_1 = 0 \wedge p_2 = 0 \wedge p_3 = 0 \models p = 0 \qquad \cdots(A)$

Example: Consider a solution

$$\langle \underline{c_1}, \ldots, \underline{c_5} \rangle = \langle 1, \ -1, \ 0, \ 0, \ 0 \rangle$$

Then,

$$p_1 = 0 \wedge p_2 = 0 \wedge p_3 = 0 \models \underbrace{1}_{c_1} x^2 + \underbrace{-1}_{c_2} y^2 + \underbrace{0}_{c_3} z^2 + \cdots + \underbrace{0}_{c_5} = 0$$

$$\boxed{x^2 - y^2 = 0}$$

3. Encode the conditions: Initiation

The condition

$$\Theta \models p = 0$$

is encoded by reducing p wrt to the Gröbner basis G of $\{\Theta\}$:

$$p \xrightarrow{G} \cdots \xrightarrow{G} NF(p)$$

and setting

$$NF(p) \equiv 0$$

which produces a set S_0 of linear constraints on $\{c_1, \ldots c_{10}\}$.

3. Encode the conditions: Consecution

The condition

$$p = 0 \ \wedge \ \rho_{\tau_i} \ \models \ p' = 0$$

is not practical to encode. Instead we encode one of

$$\rho_{\tau_i} \ \models \ p' = 0$$
$$\rho_{\tau_i} \ \models \ p' - p = 0$$

which result in a set S_i of linear constraints,
or more general

$$\exists \text{ real } \lambda. \quad \rho_{\tau_i} \ \models \ p' - \lambda p = 0$$
$$\exists \text{ polynomial } q. \quad \rho_{\tau_i} \ \models \ p' - qp = 0$$

which result in a set of nonlinear constraints.

4. Solve the constraints

Solve

$$S_0 \ \wedge \ S_1 \wedge \ldots \wedge S_k$$

for $\{c_1, \ldots, c_{10}\}$

5. Solutions

For all solutions of $\{c_1, \ldots, c_{10}\}$,

$$c_1x^3 + c_2x^2y + c_3x^2z + c_4xy^2 + c_5xyz + c_6xz^2 + c_7y^3 + c_8y^2z + c_9yz^2 + c_{10}z^3 = 0$$

is an invariant.

- **Good news**:
  Constraints are all linear: can be solved efficiently

- **Bad news**:
  Invariants are missed because of strengthening the conditions

  Trade-off between complexity and generality

Example: Nonlinear Invariant Generation

$$\textbf{integer } i, j, k, s \textbf{ where } (s = 0 \ \wedge \ j = k \ \wedge \ j \geq 0)$$

l_0 : **while** $(k \geq 0)$ **do**

  l_1 : $(s, k) := (s + i, k - 1)$

l_2 :

Target Invariant: $\boxed{p = \underline{c_1} s + \underline{c_2} ik + \underline{c_3} ij + \underline{c_4} jk + \underline{c_5}}$

Question: For what values of $\underline{c_1}, \ldots, \underline{c_5}$, is $p = 0$ inductive at l_0?

Example: Nonlinear Invariant Generation

1. Fix a template (usually a "generic polynomial" of degree m),

$$\underline{c_1}s + \underline{c_2}ik + \underline{c_3}ij + \underline{c_4}jk + \underline{c_5}$$

2. Generate constraints by encoding initiation and consecution,

$$\underline{c_2} + \underline{c_3} = 0 \ \wedge \ \underline{c_4} = \underline{c_5} = 0 \ \wedge \ \underline{c_1} - \underline{c_2} = 0$$

3. Solve the constraints,

$$\underline{c_1} = \underline{c_2} = -1, \ \underline{c_3} = 1, \ \underline{c_4} = \underline{c_5} = 0$$

4. Generate the invariant

$$-s - ik + ij = 0$$

Invariant: $\boxed{s = i(j-k)}$ at l_0.

Example: Back to GCD-LCM

$$\textbf{integer } x_1, x_2, y_1, y_2, y_3, y_4 \textbf{ where}$$
$$\left(\; y_1 = x_1 \; \wedge \; y_2 = y_3 = x_2 \; \wedge \; y_4 = 0 \; \right)$$
$$l_0 : \; \textbf{while } (y_1 \neq y_2) \textbf{ do}$$
$$\left[\begin{array}{l} l_1 : \; \textbf{while } (y_1 > y_2) \textbf{ do} \\ \qquad (y_1, y_4) := (y_1 - y_2, y_4 + y_3) \\ l_2 : \; \textbf{while } (y_2 > y_1) \textbf{ do} \\ \qquad (y_2, y_3) := (y_2 - y_1, y_3 + y_4) \end{array} \right]$$
$$\{ y_1 = \; \mathrm{GCD}(x_1, x_2), \; y_3 + y_4 = \mathrm{LCM}(x_1, x_2) \}$$

Example

1. Fix a template (usually a "generic polynomial" of degree m),

$$\underline{c_1}x_1x_2 + \underline{c_2}y_1x_2 + \underline{c_3}y_2x_1 + \underline{c_4}y_1y_3 + \underline{c_5}y_2y_4 + \underline{c_6} = 0$$

2. Generate constraints by encoding initiation and consecution,

$$\underline{c_1} + \underline{c_2} + \underline{c_3} + \underline{c_4} = 0 \ \wedge \ \underline{c_6} = 0 \quad \cdots \text{ Initiation}$$
$$\underline{c_4} = \underline{c_5} \ \wedge \ \underline{c_2} = 0 \quad \cdots \text{ Consecution } \tau_1$$
$$\underline{c_4} = \underline{c_5} \ \wedge \ \underline{c_3} = 0 \quad \cdots \text{ Consecution } \tau_2$$

3. Solve the constraints,

$$\langle \underline{c_1}, \ \underline{c_2}, \ \underline{c_3}, \ \underline{c_4}, \ \underline{c_5}, \ \underline{c_6} \rangle = \langle -1, \ 0, \ 0, \ 1, \ 1, \ 0 \rangle$$

4. Generate the invariant at ℓ_0 : $-x_1x_2 + y_1y_3 + y_2y_4 = 0$,

$$\boxed{y_1y_3 + y_2y_4 = x_1x_2}$$

Example

$$\textbf{integer } x_1, x_2, y_1, y_2, y_3, y_4 \textbf{ where } \cdots$$

$$\ell_0 : \textbf{ while } (y_1 \neq y_2) \textbf{ do} \qquad \cdots \boxed{\begin{array}{l} y_1 y_3 + y_2 y_4 = x_1 x_2 \wedge \\ \mathrm{GCD}(y_1, y_2) = \mathrm{GCD}(x_1, x_2) \end{array}}$$

$$\left[\begin{array}{l} \ell_1 : \textbf{ while } (y_1 > y_2) \textbf{ do} \\ \quad (y_1, y_4) := (y_1 - y_2, y_4 + y_3) \\ \ell_2 : \textbf{ while } (y_2 > y_1) \textbf{ do} \\ \quad (y_2, y_3) := (y_2 - y_1, y_3 + y_4) \end{array}\right]$$

$$\{y_1 = \mathrm{GCD}(x_1, x_2),\ y_3 + y_4 = \mathrm{LCM}(x_1, x_2)\}$$

Proving Partial Correctness:
Discovered Assertion + “GCD facts” suffice!


LELAND STANFORD JUNIOR UNIVERSITY
DIE LUFT DER FREIHEIT WEHT
1891

Summary

Advantages of Constraint-based Approach

- Controlling the complexity of the constraints
  - Strengthen the conditions on the property

$$\begin{array}{rcl} \Theta & \models & \psi \\ \psi \wedge \rho_\tau & \models & \psi' \end{array} \quad \Longrightarrow \quad \begin{array}{rcl} \Theta & \models & \psi \\ \rho_\tau & \models & \psi' \end{array}$$

parametric linear constraints linear constraints

  - Constrain the property

$$c_1x^3 + c_2x^2y + c_3x^2z + c_4xy^2 + c_5xyz + c_6xz^2 + c_7y^3 + c_8y^2z + c_9yz^2 + c_{10}z^3$$

$$\Downarrow$$

$$c_1x^3 + c_2xy^2 + c_3xz^2 + c_4y^2z$$

Constraint-based Approach (Cont)

Advangtages:

- Not limited to invariants
  - termination
  - temporal properties (LTL safety)
- Can exploit system structure to simplify the constraint system
  - Petri nets
- Can take advantage of results in constraint solving community:
  - Sophisticated techniques from linear algebra and algebraic geometry
  - Exploits recent advances

Disadvantages:

- Hard to solve constraints exactly.
- Domain is fixed (*e.g.*, fixed degree bound).

Papers

- Termination analysis (TACAS'01, CAV'02, CAV'05)
- Linear invariant generation (CAV'03, SAS'04, VMCAI'05, VMCAI'06)
- Nonlinear invariant generation (POPL'04)
- Nonlinear invariant generation for hybrid systems (HSCC'04)
- Differential equations (HSCC'06)

Sriram Sankaranarayanan, Mathematical Analysis of Programs, PhD Thesis, Stanford, 2005.

Related Work

- Abstract interpretation
  - - [Cousot,Cousot'77]
  - - [Cousot,Halbwachs'79]
- Set-constraint based analysis
  - - [Heintze'93]
  - - [Aiken'99]
- Termination analysis
  - - [Podelski,Rybalchenko,VMCAI'04,LICS'04]
  - - [Cousot, VMCAI'05]
- Nonlinear invariants
  - - [Bensalem et al, SAS'00]
  - - [Müller-Olm,Seidl,SAS'02,POPL'04]
  - - [Tiwari et al, TACAS'01,HSCC'03]
  - - [Rodriguez-Carbonell,Kapur,ISSAC'04]
  - - [Cousot, VMCAI'05]

Current Topics of Investigation

- Classification of systems with simpler constraint systems
- Extension to game properties (ATL*)
- Extension to other domains, in particular nonlinear inequalities
- More efficient constraint solving strategies