Logico-Numerical Max-Strategy Iteration

Peter Schrammel and Pavle Subotic
peter.schrammel@inria.fr, pavle.subotic@it.uu.se

INRIA Grenoble – Rhône-Alpes, France
Uppsala University, Sweden

COST Action Meeting in Haifa, IL

Reachability Analysis Using Abstract Interpretation

Reachability Analysis

- Solve $S = \underbrace{S_0 \cup \mathrm{post}}_{F}(S)$
- Not computable in the general case

Classical Abstract Interpretation

- Solve $S = F(S)$ in an abstract domain over-approximating the concrete reachable set
- Use an extrapolation operator ("widening") to guarantee termination: induces hard-to-predict approximations

Strategy Iteration

- Solve a sequence of "simpler" fixed point equations: $S = F^{(i)}(S)$
- Guaranteed to converge to the global least fixed point $S = F(S)$ in a finite number of steps
- Limited to Numerical domains via template polyhedra

Difficulty of Boolean Variables

- Boolean and Numerical values tightly interact.
- Classical approach: Enumerating the boolean state space - perform numerical analysis on the obtained CFG
  - State space explosion $\rightarrow$ intractable for larger programs

Implicit approaches Boolean Variables

Booleans as *integers* $\in \{0,1\}$:

- Use max-strategy iteration "as is"
- Only convex constraints $\rightarrow$ very bad precision on Booleans

Logico-numerical abstract domains (Bultan et al 1997, Jeannet et al 1999, Blanchet et al 2003):

- Logico numerical state sets $\in \wp(\mathbb{B}^m \times \mathbb{R}^n)$ abstracted by a logico numerical state abstract value
- Usually combine BDDs and numerical abstract domains

Our approach: logico-numerical abstract domains

Outline

Template Polyhedra (Sankaranarayanan et al 2005)

Polyhedra with a shape fixed by a template $\mathbf{T} \in \mathbb{R}^{m \times n}$
Generates polyhedra $\{\boldsymbol{x} \mid \boldsymbol{x} \in \mathbb{R}^n, \mathbf{T}\boldsymbol{x} \leq \boldsymbol{d}\}$ for $\boldsymbol{d} \in \overline{\mathbb{R}}^m$ $(\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\})$

Example: Intervals

Template $\mathbf{T} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ for a program with a single variable x:
template polyhedra $\begin{pmatrix} 1 \\ -1 \end{pmatrix} x \leq \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$, *i.e.*, $-d_2 \leq x \leq d_1$.

Abstract value: represented by the vector of bounds $\boldsymbol{d}$ ($\top = \infty$ and $\bot = -\infty$)

Operations: performed efficiently with the help of linear programming

Reachability analysis: find the smallest bounds representing a fixed point of the semantic equations

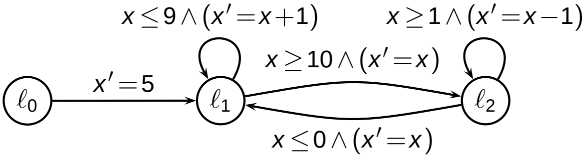
Numerical Max-Strategy Iteration (Gawlitza and Seidl 2007)

General idea: Compute the least fixed point of the semantic equation system $\mathcal{M}$ by:

- computing a sequence of $lfp[\![\mu]\!]$ using linear programming
- until $lfp[\![\mu]\!] = lfp[\![\mathcal{M}]\!]$

- A strategy μ choses exactly one argument on the right-hand side of each equation.
- We let $\delta_{l,t}$ is the bound value for a location l and template bound t.

Example: Strategy



$$\delta_{0,1} = \infty$$

$$\delta_{0,2} = \infty$$

$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\left\{x' \mid x \leq \delta_{0,1} \wedge x' = 5\right\} \\ \sup\left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \wedge x \leq 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \wedge x \leq 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\left\{-x' \mid -x \leq \delta_{0,2} \wedge x' = 5\right\} \\ \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \wedge x \leq 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \wedge x \leq 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \wedge x \geq 10 \\ \wedge\, x' = x \end{array} \right\} , \quad \sup\left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \wedge x \geq 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \wedge x \leq 10 \\ \wedge\, x' = x \end{array} \right\}, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \wedge x \geq 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

Numerical Max-Strategy Iteration (Gawlitza and Seidl 2007)

General idea: Compute the least fixed point of the semantic equation system $\mathcal{M}$ by:

- computing a sequence of $lfp[\![\mu]\!]$ using linear programming
- until $lfp[\![\mu]\!] = lfp[\![\mathcal{M}]\!]$

A strategy μ choses exactly one argument on the right-hand side of each equation.

A strategy μ' is called an improvement of μ w.r.t the abstract value ***d*** iff

1. it is "at least as good" as μ with respect to *d* and
2. it is "strictly better for the changed equations"

Max-Strategy Improvement Algorithm

initial strategy: $\mu := \{\delta_{\ell_0} \geq \infty,\ \delta_\ell \geq -\infty \text{ for all } \ell \neq \ell_0\}$

initial abstract value: $\boldsymbol{d} := \lambda\ell.\delta_\ell \to \begin{cases} \infty & \text{for } \ell = \ell_0 \\ -\infty & \text{for } \ell \neq \ell_0 \end{cases}$

while not ***d*** is a solution of $\mathcal{M}$ do

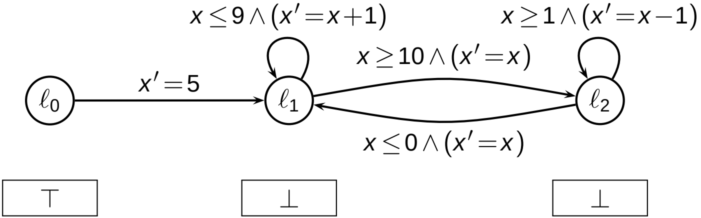
  $\mu :=$ improvement of μ w.r.t ***d***

  d $:= lfp[\![\mu]\!]$

done

return ***d***

Example



$$\delta_0 = \infty$$

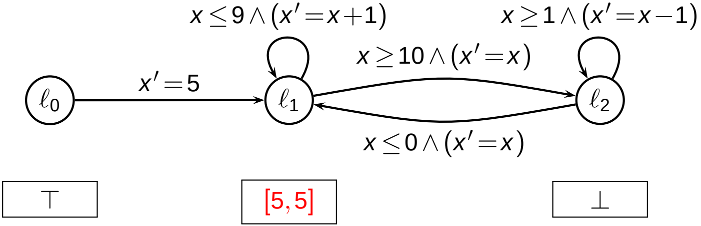
$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\left\{x' \mid x \le \delta_{0,1} \wedge x' = 5\right\} \\ \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \le 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \le 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\left\{-x' \mid -x \le \delta_{0,2} \wedge x' = 5\right\} \\ \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \le 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \ge 10 \\ \wedge\, x' = x \end{array} \right\}, \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \ge 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 10 \\ \wedge\, x' = x \end{array} \right\}, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \ge 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

Example



$\delta_0 = \infty$

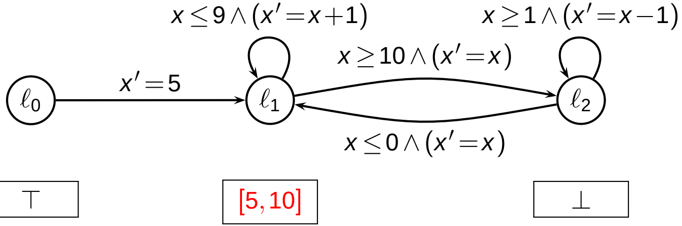
$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup \{x' \mid x \leq \delta_{0,1} \ \wedge \ x' = 5\} \\ \sup \left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \ \wedge \ x \leq 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup \left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \ \wedge \ x \leq 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup \{-x' \mid -x \leq \delta_{0,2} \ \wedge \ x' = 5\} \\ \sup \left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \ \wedge \ x \leq 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup \left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \ \wedge \ x \leq 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup \left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \ \wedge \ x \geq 10 \\ \wedge\, x' = x \end{array} \right\}, \sup \left\{ x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \ \wedge \ x \geq 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup \left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \ \wedge \ x \leq 10 \\ \wedge\, x' = x \end{array} \right\}, \sup \left\{ -x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \ \wedge \ x \geq 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

Example



$$\delta_0 = \infty$$

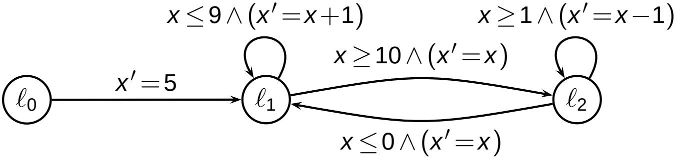
$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{x' \mid x \le \delta_{0,1} \wedge x' = 5\} \\ \sup\left\{x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \le 9 \\ \wedge x' = x+1 \end{array}\right\} & , \quad \sup\left\{x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \le 0 \\ \wedge x' = x \end{array}\right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{-x' \mid -x \le \delta_{0,2} \wedge x' = 5\} \\ \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 9 \\ \wedge x' = x+1 \end{array}\right\} & , \quad \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \le 0 \\ \wedge x' = x \end{array}\right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \ge 10 \\ \wedge x' = x \end{array}\right\}, \sup\left\{x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \ge 1 \\ \wedge x' = x-1 \end{array}\right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 10 \\ \wedge x' = x \end{array}\right\}, \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \ge 1 \\ \wedge x' = x-1 \end{array}\right\} \right\}$$

Example



$\boxed{\top}$ $\boxed{[5,10]}$ $\boxed{[10,10]}$

$$\delta_0 = \infty$$

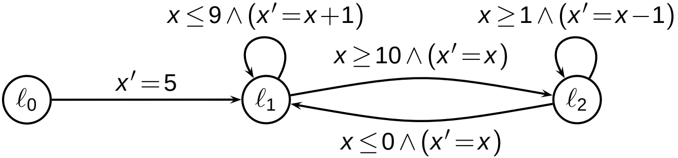
$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{x' \mid x \le \delta_{0,1} \wedge x' = 5\} \\ \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \le 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \le 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{-x' \mid -x \le \delta_{0,2} \wedge x' = 5\} \\ \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 9 \\ \wedge\, x' = x+1 \end{array} \right\} & , \quad \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \le 0 \\ \wedge\, x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{1,1} \wedge x \ge 10 \\ \wedge\, x' = x \end{array} \right\}, \sup\left\{ x' \,\middle|\, \begin{array}{c} x \le \delta_{2,1} \wedge x \ge 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{1,2} \wedge x \le 10 \\ \wedge\, x' = x \end{array} \right\}, \sup\left\{ -x' \,\middle|\, \begin{array}{c} -x \le \delta_{2,2} \wedge x \ge 1 \\ \wedge\, x' = x-1 \end{array} \right\} \right\}$$

Example



$\top$ | $[5, 10]$ | $[0, 10]$

$$\delta_0 = \infty$$

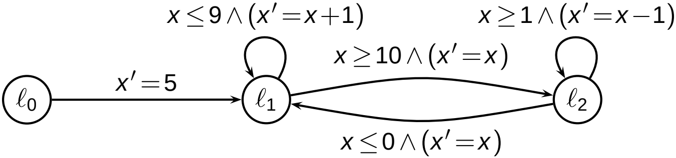
$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty, & \sup\{x' \mid x \leq \delta_{0,1} \wedge x' = 5\} \\ \sup\left\{x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \wedge x \leq 9 \\ \wedge\, x' = x+1 \end{array}\right\}, & \sup\left\{x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \wedge x \leq 0 \\ \wedge\, x' = x \end{array}\right\} \end{array} \right\}$$

$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty, & \sup\{-x' \mid -x \leq \delta_{0,2} \wedge x' = 5\} \\ \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \wedge x \leq 9 \\ \wedge\, x' = x+1 \end{array}\right\}, & \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \wedge x \leq 0 \\ \wedge\, x' = x \end{array}\right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{x' \,\middle|\, \begin{array}{c} x \leq \delta_{1,1} \wedge x \geq 10 \\ \wedge\, x' = x \end{array}\right\}, \sup\left\{x' \,\middle|\, \begin{array}{c} x \leq \delta_{2,1} \wedge x \geq 1 \\ \wedge\, x' = x-1 \end{array}\right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \leq \delta_{1,2} \wedge x \leq 10 \\ \wedge\, x' = x \end{array}\right\}, \sup\left\{-x' \,\middle|\, \begin{array}{c} -x \leq \delta_{2,2} \wedge x \geq 1 \\ \wedge\, x' = x-1 \end{array}\right\} \right\}$$

Example



$\boxed{\top}$ $\boxed{[0,10]}$ $\boxed{[0,10]}$

$$\delta_0 = \infty$$

$$\delta_{1,1} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{x' \mid x \le \delta_{0,1} \ \wedge\ x' = 5\} \\ \sup\left\{ x' \middle| \begin{array}{c} x \le \delta_{1,1} \ \wedge\ x \le 9 \\ \wedge\ x' = x+1 \end{array} \right\} & , \quad \sup\left\{ x' \middle| \begin{array}{c} x \le \delta_{2,1} \ \wedge\ x \le 0 \\ \wedge\ x' = x \end{array} \right\} \end{array} \right\}$$

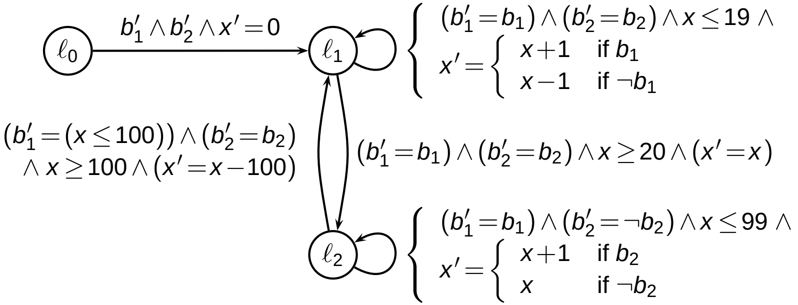
$$\delta_{1,2} = \bigsqcup \left\{ \begin{array}{ll} -\infty & , \quad \sup\{-x' \mid -x \le \delta_{0,2} \ \wedge\ x' = 5\} \\ \sup\left\{ -x' \middle| \begin{array}{c} -x \le \delta_{1,2} \ \wedge\ x \le 9 \\ \wedge\ x' = x+1 \end{array} \right\} & , \quad \sup\left\{ -x' \middle| \begin{array}{c} -x \le \delta_{2,2} \ \wedge\ x \le 0 \\ \wedge\ x' = x \end{array} \right\} \end{array} \right\}$$

$$\delta_{2,1} = \bigsqcup \left\{ -\infty, \sup\left\{ x' \middle| \begin{array}{c} x \le \delta_{1,1} \ \wedge\ x \ge 10 \\ \wedge\ x' = x \end{array} \right\}, \quad \sup\left\{ x' \middle| \begin{array}{c} x \le \delta_{2,1} \ \wedge\ x \ge 1 \\ \wedge\ x' = x-1 \end{array} \right\} \right\}$$

$$\delta_{2,2} = \bigsqcup \left\{ -\infty, \sup\left\{ -x' \middle| \begin{array}{c} -x \le \delta_{1,2} \ \wedge\ x \le 10 \\ \wedge\ x' = x \end{array} \right\}, \sup\left\{ -x' \middle| \begin{array}{c} -x \le \delta_{2,2} \ \wedge\ x \ge 1 \\ \wedge\ x' = x-1 \end{array} \right\} \right\}$$

Logico-Numerical Programs

```
b1=b2=true;
x=0;
while(true) {
  while(x<=19) { x = b1 ? x+1 : x-1; }
  while(x<=99) { x = b2 ? x+1 : x; b2 = !b2; }
  if (x>=100) { b1 = (x<=100); x = x-100; }
}
```



Abstract Domain

$$\wp(\mathbb{B}^p \times \mathbb{R}^n) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \wp(\mathbb{B}^p) \times \overline{\mathbb{R}}^m$$

Abstract value $S = (B, \boldsymbol{d})$: cartesian product of

- Valuations of the Boolean variables B
  (represented as Boolean formulas using BDDs) and
- Template bounds $\boldsymbol{d}$

Abstract domain over CFG: $Loc \to \wp(\mathbb{B}^p) \times \overline{\mathbb{R}}^m$

The Idea

1. Perform Kleene iteration until
   - for all locations the set of reachable Boolean states does not change no matter what transition we take.
   - We call this a subsystem, boolean state states the same but numerical state evolves
2. Continue Kleene iteration when Numerical values make us leave the subsystem
3. Solution when system wide numerical and boolean values stable

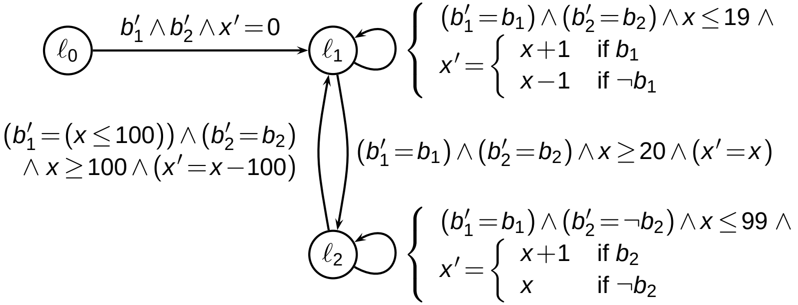
Algorithm

$S := S^0$
$S' = \mathrm{post}(S)$
while $S \neq S'$ do
  while $B \neq B'$ do
    $S := S'$
    $S' = \mathrm{post}(S)$
  done
  $S := S'$
  $\mathcal{M} = \mathrm{generate}(S)$
  $\mu := (\delta = \boldsymbol{d})$
  $\mu' = \mathrm{max\_improve}(\mu, \boldsymbol{d})$
  while $\mu' \neq \mu$ do
    $\mu := \mu'$
    $\boldsymbol{d} := \mathit{lfp}[\![\mu]\!]$
    $\mu' = \mathrm{max\_improve}(\mu, \boldsymbol{d})$
  done
  $S' = \mathrm{post}(S)$
done
return S

phase (1): truncated logico-numerical Kleene iteration

phase (2): numerical max-strategy iteration

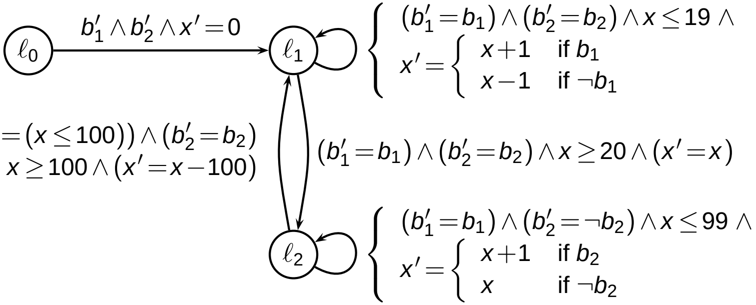
Example



Interval template: $\begin{pmatrix} 1 \\ -1 \end{pmatrix}$

Notation: $\begin{pmatrix} \varphi(b_1, b_2) \\ [-\delta_{\ell,2}, \delta_{\ell,1}] \end{pmatrix}$

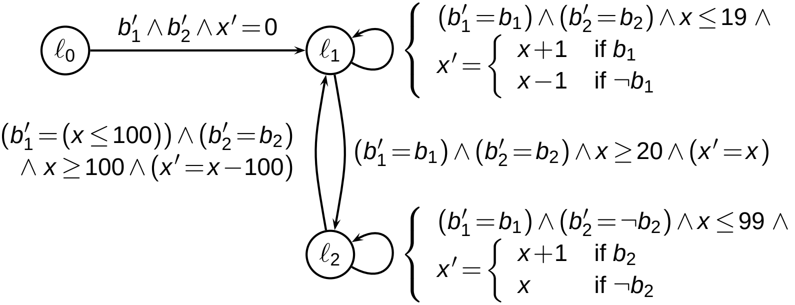
Example



Initial state:

ℓ_0	ℓ_1	ℓ_2
$\top$	$\bot$	$\bot$

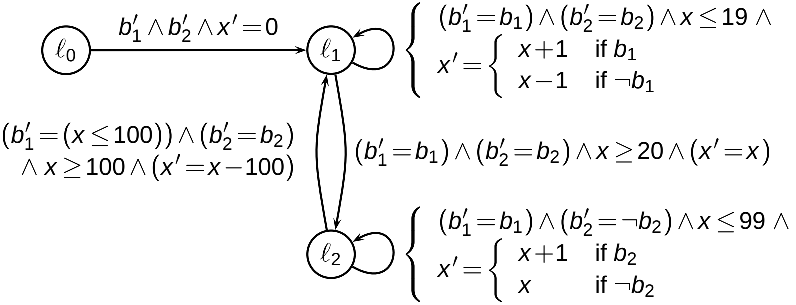
Example



Phase (1): propagation through (ℓ_0, R, ℓ_1):

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0,0] \end{pmatrix}$	$\bot$

Example

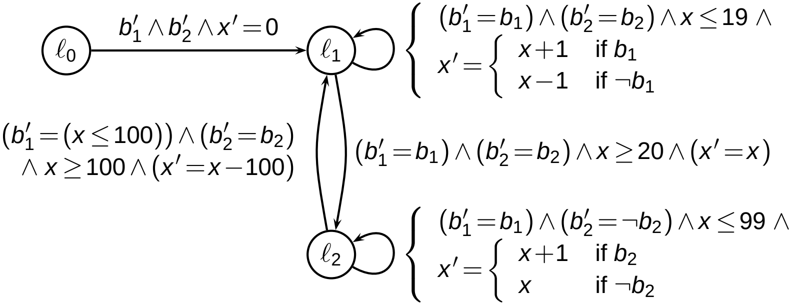


Phase (1): propagation through (ℓ_1, R, ℓ_1):

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0,1] \end{pmatrix}$	$\bot$

(preliminarily stable)

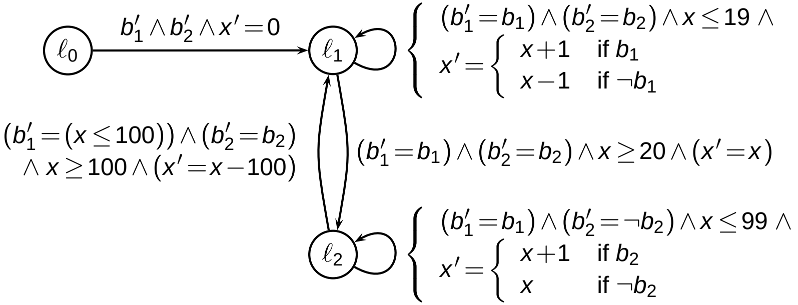
Example



Phase (2): generate equation system:

$$
\begin{array}{l}
\delta_0 = \infty \\
\delta_{1,1} = \bigsqcup \{1, \sup\{x' \mid x' = 0\}, \sup\{x' \mid x \leq \delta_{1,1} \wedge x' = x+1 \wedge x \leq 19\}\} \\
\delta_{1,2} = \bigsqcup \{0, \sup\{-x' \mid x' = 0\}, \sup\{-x' \mid -x \leq \delta_{1,2} \wedge x' = x+1 \wedge x \leq 19\}\} \\
\delta_2 = -\infty
\end{array}
$$

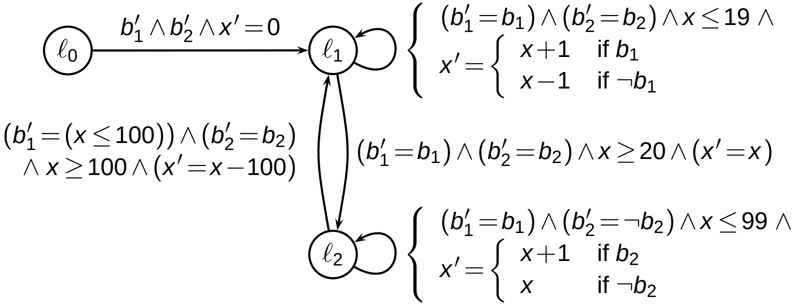
Example



Phase (2): initial strategy:

$$\begin{aligned}
&\delta_0 = \infty \\
&\delta_{1,1} = \bigsqcup\{1, \sup\{x' \mid x' = 0\}, \sup\{x' \mid x \leq \delta_{1,1} \wedge x' = x+1 \wedge x \leq 19\}\} \\
&\delta_{1,2} = \bigsqcup\{0, \sup\{-x' \mid x' = 0\}, \sup\{-x' \mid -x \leq \delta_{1,2} \wedge x' = x+1 \wedge x \leq 19\}\} \\
&\delta_2 = -\infty
\end{aligned}$$

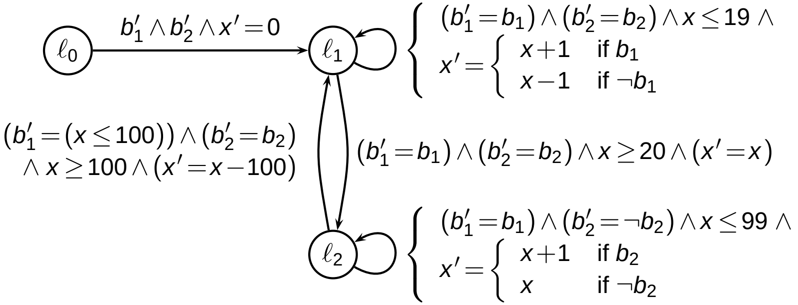
Example



Phase (2): improve strategy w.r.t. $\delta_{1,1}$:

$$\begin{aligned}
&\delta_0 = \infty \\
&\delta_{1,1} = \bigsqcup\{1, \sup\{x' \mid x'=0\}, \sup\{x' \mid x \leq \delta_{1,1} \wedge x'=x+1 \wedge x\leq 19\}\} \\
&\delta_{1,2} = \bigsqcup\{0, \sup\{-x' \mid x'=0\}, \sup\{-x' \mid -x \leq \delta_{1,2} \wedge x'=x+1 \wedge x\leq 19\}\} \\
&\delta_2 = -\infty
\end{aligned}$$

Example

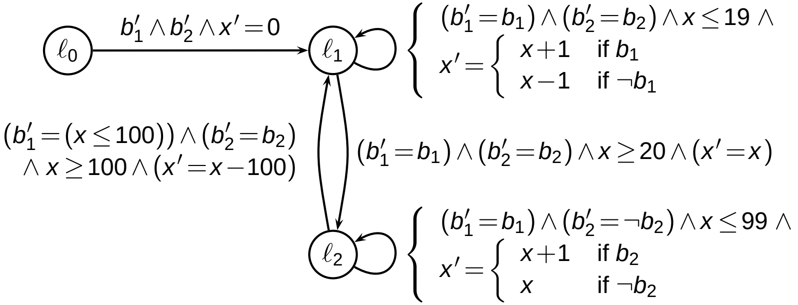


Phase (2): fixed point:

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0, 20] \end{pmatrix}$	$\perp$

(no more improvement)

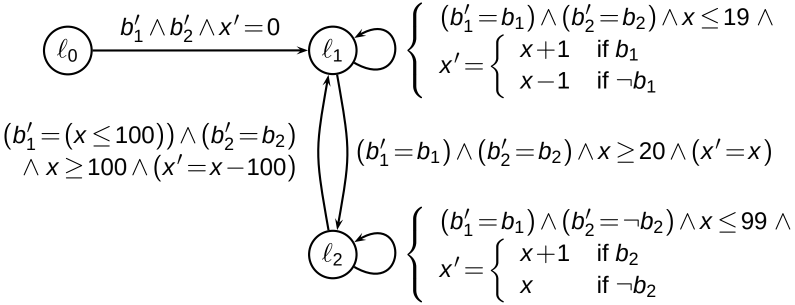
Example



Phase (1): propagation through (ℓ_1, R, ℓ_2):

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0,20] \end{pmatrix}$	$\begin{pmatrix} b_1 \wedge b_2 \\ [20,20] \end{pmatrix}$

Example

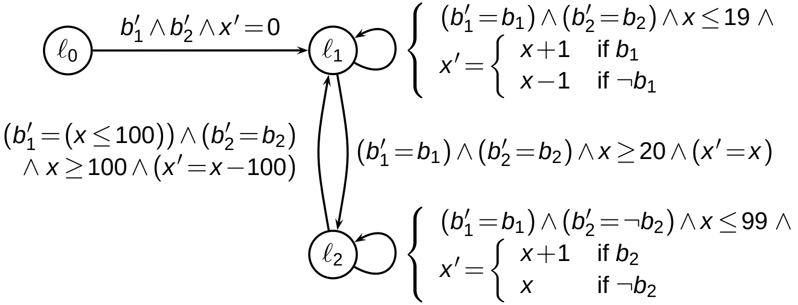


Phase (1): propagation through (ℓ_2, R, ℓ_2):

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0,20] \end{pmatrix}$	$\begin{pmatrix} b_1 \\ [20,21] \end{pmatrix}$

(preliminarily stable)

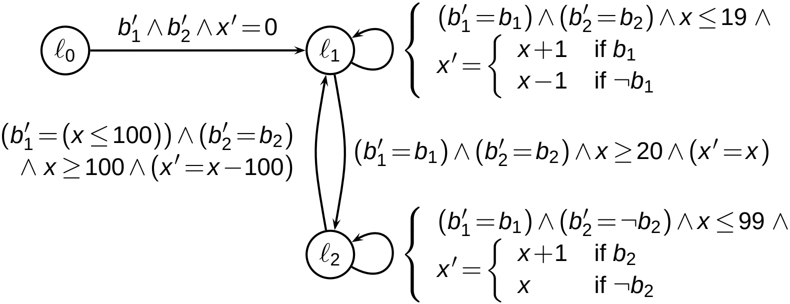
Example



Phase (2): generate equation system:

$$
\begin{array}{llcll}
\delta_0 = \infty & & & & \\
\delta_{1,1} = \bigsqcup\{ & 20 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{1,2} = \bigsqcup\{ & 0 & , & \sup\{-x' \mid x \leq \delta_{2,2} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{2,1} = \bigsqcup\{ & 21 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\} \\
\delta_{2,2} = \bigsqcup\{ & -20 & , & \sup\{-x' \mid -x \leq \delta_{2,2} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\}
\end{array}
$$

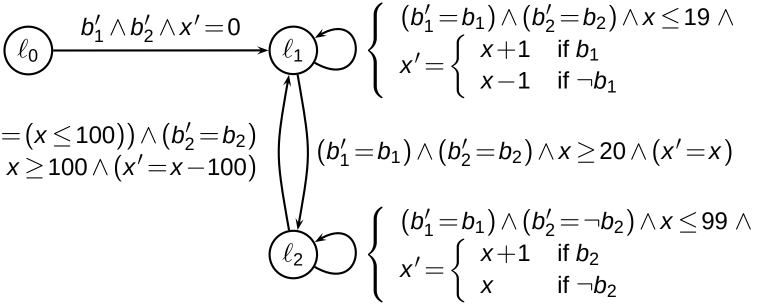
Example



Phase (2): initial strategy:

$$
\begin{array}{lrcll}
\delta_0 = \infty & & & & \\
\delta_{1,1} = \bigsqcup\{ & 20 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{1,2} = \bigsqcup\{ & 0 & , & \sup\{-x' \mid x \leq \delta_{2,2} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{2,1} = \bigsqcup\{ & 21 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\} \\
\delta_{2,2} = \bigsqcup\{ & -20 & , & \sup\{-x' \mid -x \leq \delta_{2,2} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\}
\end{array}
$$

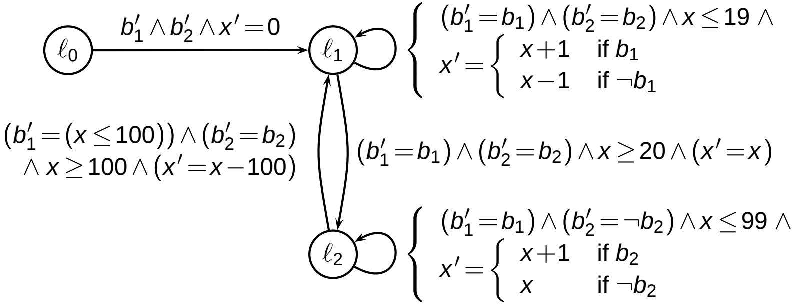
Example



Phase (2): improvement w.r.t. $\delta_{2,1}$:

$$
\begin{array}{lrcll}
\delta_0 = \infty & & & & \\
\delta_{1,1} = \bigsqcup\{ & 20 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{1,2} = \bigsqcup\{ & 0 & , & \sup\{-x' \mid x \leq \delta_{2,2} \wedge x' = x - 100 \wedge x \geq 100\} & , \ldots\} \\
\delta_{2,1} = \bigsqcup\{ & 21 & , & \sup\{x' \mid x \leq \delta_{2,1} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\} \\
\delta_{2,2} = \bigsqcup\{ & -20 & , & \sup\{-x' \mid -x \leq \delta_{2,2} \wedge x' = x + 1 \wedge x \leq 99\} & , \ldots\}
\end{array}
$$

Example

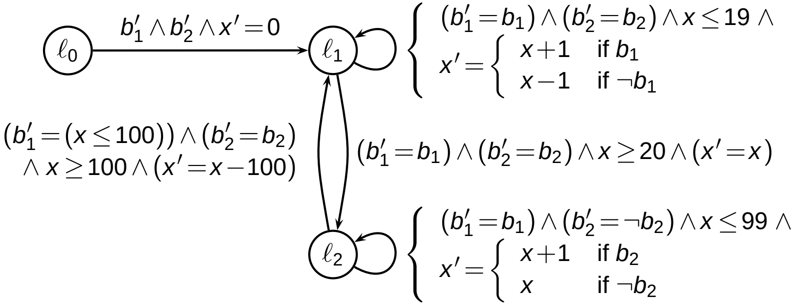


Phase (2): fixed point:

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \wedge b_2 \\ [0, 20] \end{pmatrix}$	$\begin{pmatrix} b_1 \\ [20, 100] \end{pmatrix}$

(no more improvement)

Example



Phase (1): propagation through (ℓ_2, R, ℓ_1):

ℓ_0	ℓ_1	ℓ_2
$\top$	$\begin{pmatrix} b_1 \\ [0,20] \end{pmatrix}$	$\begin{pmatrix} b_1 \\ [20,100] \end{pmatrix}$

(global fixed point)

Properties

The logico-numerical max-strategy algorithm

- terminates after a finite number of iterations (termination).
- computes a fixed point of the semantic equations (soundness).
- computes the least fixed point of the semantic equations (optimality).

Experiments

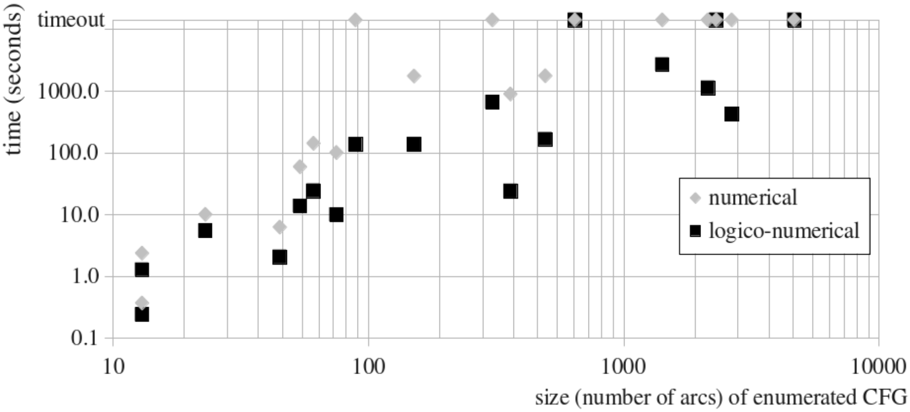
High-level simulation models (programmed in LUSTRE) of manufacturing systems

- Consist of building blocks like sources, buffers, machines and routers that synchronize via handshakes.
- Produce enumerated CFGs of up to 650 locations and 5000 transitions after simplification by Boolean reachability

Comparing the precision of the inferred invariants of

- Numerical max-strategy iteration (MSI) on the enumerated CFG
- Logico-numerical max-strategy iteration (LNMSI) on CFG obtained by state space partitioning by “discrete numerical modes”: equivalence classes of Boolean valuations implying the same numerical transitions relations

Results


time (seconds)
timeout
1000.0
100.0
10.0
1.0
0.1
10
100
1000
10000
size (number of arcs) of enumerated CFG
numerical
logico-numerical

Results

- LNMSI scales better than MSI: 9 times faster – for those benchmarks where both methods terminated before the timeout: MSI hit the timeout in 8 out of 18 cases (versus 3 for LNMSI)
- *Precision* is *almost preserved* to 100%, due to the better scalability even able to prove 3 more benchmarks.
- Gain in speed increases with the template size.

Comparison with logico-numerical analysis with octagons using the standard approach with widening:

- 18% of the bounds strictly better with LNMSI: in 2 cases these improvements made the difference to prove the property.
- Standard analysis 19 times faster on average

Future Work

Future work:

- Tackle efficiency issues by designing a more integrated logico-numerical max-strategy solver.
- Apply our method to the analysis of logico-numerical hybrid automata (Schrammel and Jeannet 2012) by extending hybrid max-strategy iteration (Dang and Gawlitza 2011)