

Linear absolute value relation analysis

Liqian Chen¹ Antoine Miné^{2,3} Ji Wang¹ Patrick Cousot^{2,4}

¹National Lab. for Parallel and Distributed Processing, Changsha, China

²École Normale Supérieure, Paris, France

³CNRS, France

⁴CIMS, New York University, New York, NY, USA

30/03/2011 – ESOP 2011

Overview

- Motivation
- Double description method for linear absolute value systems
- An abstract domain of linear absolute value inequalities
- Implementation and Experiments
- Conclusion

Motivation

Numerical static analysis by abstract interpretation

Numerical static analysis

- discover **numerical** properties of a program **statically** and **automatically**

Theoretical framework: **abstract interpretation**

to design static analyses that are

- **sound** by construction (no behavior is omitted)
- **approximate** (trade-off between precision and efficiency)

Numerical abstract domains

- infer relationships among numerical variables
- examples
 - Intervals ($a \leq x \leq b$), Octagons ($\pm x \pm y \leq c$), Polyhedra ($\sum_k a_k x_k \leq b$)

Polyhedra and Sub-polyhedra abstract domains

The polyhedra abstract domain [Cousot Halbwachs 78]

- linear relation analysis to infer linear invariants

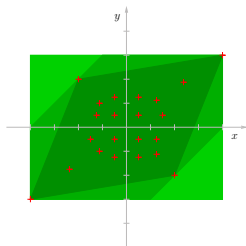
$$\bigwedge \sum_i a_i x_i \leq b$$

where $a_i, b \in \mathbb{I}$ and $\mathbb{I} \in \{\mathbb{Q}, \mathbb{R}\}$

- implementations
 - Polylib, NewPolka (in APRON), PPL, ...

Sub-polyhedra abstract domains

- octagons ($\pm x \pm y \leq c$) [Miné 01]
- octahedra ($\sum_i \pm x_i \leq c$) [Clarísó et al. 04]
- TVPI ($ax_i + bx_j \leq c$) [Simon et al. 03]
- template polyhedra [Sankaranarayanan et al. 05]
($\sum_i a_i x_i \leq c$ where a_i are fixed beforehand)
- ...



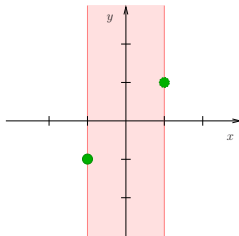
Motivation

Convexity limitations: a motivating example

```

1:   real x, y;
2:   x ← 1;
3:   y ← 1;
4:   while (true) {
5:       x ← -x;
6:       y ← 1/x; ①
7:   }

```



Loc	Most abstract domains	Concrete semantics
①	$x \in [-1, 1]$ $y \in [-\infty, +\infty]$	$(x = -1 \wedge y = -1)$ $\vee (x = 1 \wedge y = 1)$

Division-by-zero?

Safe !

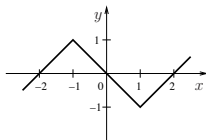
Motivation

Piecewise linear



Non-linear

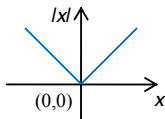
Linear



Absolute Value (AV): $y = |x|$

- piecewise linear expressiveness

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$



Possible applications

- AV functions in C: $abs()$, $fabs()$, ...
- MiniMax functions in C: $fmax()$, $fmin()$, ...
 - e.g., $\max(x, y) = \frac{1}{2}(|x - y| + x + y)$
- Abstractions for floating-point rounding errors

- $|R_{f,r}(x) - x| \leq \epsilon_{\text{rel}} \cdot |x| + \epsilon_{\text{abs}}$ (float: $\epsilon_{\text{rel}} = 2^{-23}$, $\epsilon_{\text{abs}} = 2^{-149}$)

Double Description Method for AVI systems

Equivalence among itv linear, linear AVI, XLCP systems

3 kinds of equivalent relations:

- interval linear inequalities (ILI): $\sum_k [a_k, b_k]x_k \leq c$ [Chen et al. SAS'09]
- linear absolute value inequalities (AVI): $\sum_k a'_k x_k + \sum_k b'_k |x_k| \leq c'$
- extended linear complementary problem (XLCP) inequalities:

$$\sum_k a''_k x_k^+ + \sum_k b''_k x_k^- \leq c''$$

where x_k^+, x_k^- satisfy

$$x_k^+, x_k^- \geq 0 \text{ and } \sum_k x_k^+ x_k^- = 0.$$

- $x_k^+ = 0 \vee x_k^- = 0$
- $x_k = x_k^+ - x_k^-, |x_k| = x_k^+ + x_k^-$;
- $x_k^+ = \frac{1}{2}(x_k + |x_k|), x_k^- = \frac{1}{2}(|x_k| - x_k)$;

Example

AVI: $\{|x| \leq 1, -|x| \leq -1\}$, ILI: $\{x \leq 1, -x \leq 1, [-1, 1]x \leq -1\}$,
 XLCP: $\{x^+ + x^- \leq 1, -x^+ - x^- \leq -1, x^+ \geq 0, x^- \geq 0, (x^+)^T x^- = 0\}$

Double Description Method for Polyhedra

Theorem (Minkowski-Weyl Theorem)

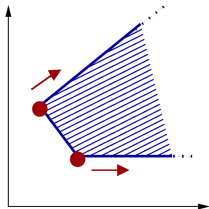
The set $P \subseteq \mathbb{R}^n$ is a polyhedron, iff it is finitely generated, i.e., there exist finite sets $V, R \in \mathbb{R}^n$ such that P can be generated by (V, R) :

$$P = \left\{ \sum_{i=1}^{|V|} \lambda_i V_i + \sum_{j=1}^{|R|} \mu_j R_j \mid \forall i, \lambda_i \geq 0, \forall j, \mu_j \geq 0, \sum_{i=1}^{|V|} \lambda_i = 1 \right\}$$

Dual representations

- constraint representation: $Ax \leq b$
 - e.g., $\{-y \leq -1, x - y \leq 1, -x - y \leq -3\}$
- generator representation: $G = (V, R)$
 - e.g., $(\{(2, 1), (1, 2)\}, \{(0, 1), (1, 1)\})$

Dual conversion: Chernikova's algorithm



XLCP: From Constraints to Generators

XLCP: $Mx^+ + Nx^- \leq c \wedge x^+, x^- \geq 0 \wedge (x^+)^T x^- = 0$

Step1: $G \leftarrow \text{Polyhedra.Cons2Gens}(Mx^+ + Nx^- \leq c \wedge x^+, x^- \geq 0)$

Step2: $G^c \leftarrow \{g \in G \mid g \text{ satisfies } (x_g^+)^T x_g^- = 0\}$

Step3: $G^{cc} \leftarrow \{ \langle G_{s_1}^c, \dots, G_{s_i}^c, \dots, G_{s_m}^c \rangle \mid G_{s_i}^c = (V_{s_i}^c, R_{s_i}^c) \text{ satisfies}$

① $V_{s_i}^c \subseteq V^c, R_{s_i}^c \subseteq R^c, \cup_{i=1}^m V_{s_i}^c = V^c, \cup_{i=1}^m R_{s_i}^c = R^c$, and

② Within each group $G_{s_i}^c$, any sum z of an arbitrary convex combination of extreme points from $V_{s_i}^c$ and an arbitrary nonnegative combination of extreme rays from $R_{s_i}^c$, satisfies the complementary condition $(z^+)^T z^- = 0$.

Theorem

Let $P_{\pm} = \{x \in \mathbb{R}^{2n} \mid Ax \geq b, x \geq 0, (x^+)^T x^- = 0\}$, and let $G^{cc} = \langle G_{s_1}^c, \dots, G_{s_i}^c, \dots, G_{s_m}^c \rangle$ be the grouping result of its complementary generators where $G_{s_i}^c = (V_{s_i}^c, R_{s_i}^c)$. Then $x \in P_{\pm}$, iff there exists some i ($i \in \mathbb{N}, 1 \leq i \leq m$) such that

$$x = \sum_{v_j^c \in V_{s_i}^c} \lambda_j v_j^c + \sum_{r_k^c \in R_{s_i}^c} \mu_k r_k^c$$

where $\lambda_j, \mu_k \geq 0, \sum_j \lambda_j = 1$.

XLCP: From Constraints to Generators (cont.)

Example

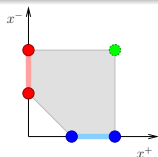
XLCP: $\{-x^+ - x^- \leq -1, x^+ \leq 2, x^- \leq 2, x^+ \geq 0, x^- \geq 0, (x^+)^T x^- = 0\}$

Polyhedral generators of $\{-x^+ - x^- \leq -1, x^+ \leq 2, x^- \leq 2, x^+ \geq 0, x^- \geq 0\}$:

$$(V, R) = \left(\begin{pmatrix} x^+ \\ x^- \end{pmatrix} : \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 \\ 2 \end{pmatrix} \right\}, \emptyset \right)$$

Grouping results of complementary generators G^{cc} :

$$\left\{ \left(\begin{pmatrix} x^+ \\ x^- \end{pmatrix} : \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix} \right\}, \emptyset \right), \left(\begin{pmatrix} x^+ \\ x^- \end{pmatrix} : \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix} \right\}, \emptyset \right) \right\}$$



XLCP: From Constraints to Generators (cont.)

XLCP: $Mx^+ + Nx^- \leq c \wedge x^+, x^- \geq 0 \wedge (x^+)^T x^- = 0$

Step1: $G \leftarrow \text{Polyhedra.Cons2Gens}(Mx^+ + Nx^- \leq c \wedge x^+, x^- \geq 0)$

Step2: $G^c \leftarrow \{g \in G \mid g \text{ satisfies } (x_g^+)^T x_g^- = 0\}$

Step3: $G^{cc} \leftarrow \{ \langle G_{s_1}^c, \dots, G_{s_i}^c, \dots, G_{s_m}^c \rangle \}$

Fortunately, when designing the AV abstract domain, we only need G^c !

XLCP: From Generators to Constraints

Step 1. $Mx^+ + Nx^- \leq b \leftarrow \text{Polyhedra.Gens2Cons}(\mathcal{G}^c);$

Step 2. add $x^+, x^- \geq 0, (x^+)^T x^- = 0$

An abstract domain of linear absolute value inequalities

The AVI abstract domain

An abstract domain of linear absolute value inequalities (AVI)

- goal: to infer linear relations among **values** and **absolute values** of program variables

$$\sum_k a_k x_k + \sum_k b_k |x_k| \leq c$$

Domain representation for domain element **P**

- representation: a linear AVI system $Ax + B|x| \leq c$
- semantics: $\gamma(\mathbf{P}) = \{x \in \mathbb{R}^n : Ax + B|x| \leq c\}$

Topological properties: can be **non-convex**, even **unconnected**

- a (possibly empty) convex polyhedron in each orthant
- e.g., $-|x| \leq -1$

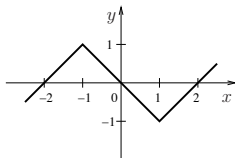
The AVI abstract domain (representation)

Expressiveness limitation: $\sum_k a_k x_k + \sum_k b_k |x_k| \leq c$

- $|\cdot|$ applies to only (single) variables rather than expressions

An example: $y = x - |x + 1| + |x - 1|$, i.e.,

$$y = \begin{cases} x + 2 & \text{if } x \leq -1 \\ -x & \text{if } -1 \leq x \leq 1 \\ x - 2 & \text{if } x \geq 1 \end{cases}$$



Expressiveness lifting

- introduce new **auxiliary variables** to denote expressions inside the AV function
- e.g., $\{y = x - |v_1| + |v_2|, v_1 = x + 1, v_2 = x - 1\}$

The AVI abstract domain (operations)

How to implement AVI domain operations for static analysis

- maintain the map between abstract environments over x and abstract environments over x^+, x^- :

$$\begin{aligned}x &= x^+ - x^-, & |x| &= x^+ + x^- \\x^+ &= \frac{1}{2}(x + |x|), & x^- &= \frac{1}{2}(|x| - x)\end{aligned}$$

where x^+, x^- satisfy $x^+ \geq 0, x^- \geq 0, (x^+)^T x^- = 0$

- let $G^c = (V^c, R^c)$ be the set of complementary generators of XLCP system:

$$\begin{aligned}Mx^+ + Nx^- &\leq b \\x^+ \geq 0, x^- \geq 0, (x^+)^T x^- &= 0\end{aligned}$$

The AVI abstract domain (operations)

Domain operations

1 lattice operations

- **emptiness test:** \mathbf{P} is empty, iff $V^c = \emptyset$
- **inclusion test:** $\mathbf{P} \sqsubseteq \mathbf{P}'$ that is $\gamma(\mathbf{P}) \subseteq \gamma(\mathbf{P}')$, iff

$$\forall v \in V^c, M' v^+ + N' v^- \leq b' \quad \wedge \quad \forall r \in R^c, M' r^+ + N' r^- \leq 0$$
- **meet:** $\mathbf{P} \sqcap \mathbf{P}'$ is an AVI domain element whose XLCP system is

$$\begin{aligned} Mx^+ + Nx^- &\leq b \\ M'x^+ + N'x^- &\leq b' \\ x^+ &\geq 0, x^- \geq 0, (x^+)^T x^- = 0 \end{aligned}$$
- **join:** $\mathbf{P} \sqcup \mathbf{P}'$ is the least AVI domain element containing \mathbf{P} and \mathbf{P}' , whose set of complementary generators is the union of those of \mathbf{P} and \mathbf{P}' : $(V^c \cup V'^c, R^c \cup R'^c)$.

The AVI abstract domain (operations)

Domain operations

② transfer functions

- **test transfer function:** $\tau \llbracket cx + d|x| \leq e \rrbracket^\sharp(\mathbf{P})$, whose XLCP system is defined as

$$\begin{aligned} Mx^+ + Nx^- &\leq b \\ (c + d)x^+ + (d - c)x^- &\leq e \\ x^+ \geq 0, x^- \geq 0, (x^+)^T x^- &= 0 \end{aligned}$$

- **projection:** $\tau \llbracket x_j := \text{random}() \rrbracket^\sharp(\mathbf{P})$, whose set of complementary generators is defined as $(V^c, R^c \cup \{e_j^+, e_j^-, -e_j^+, -e_j^-\})$, where e_j^\pm denotes a canonical basis vector
- **assignment transfer function:** $\tau \llbracket x_j := \sum_i a_i x_i + \sum_i b_i |x_i| + c \rrbracket^\sharp(\mathbf{P})$, can be implemented as:

$$(\tau \llbracket x_j := \text{random}() \rrbracket^\sharp \circ \tau \llbracket \sum_i a_i x_i + \sum_i b_i |x_i| + c - x'_j = 0 \rrbracket^\sharp(\mathbf{P})) [x'_j / x_j]$$

The AVI abstract domain (operations)

Domain operations

- ③ **widening**: given two AVI domain elements $\mathbf{P} \sqsubseteq \mathbf{P}'$, we define

$$\mathbf{P} \nabla \mathbf{P}' \stackrel{\text{def}}{=} \mathcal{S}_1 \cup \mathcal{S}_2 \cup \{x^+, x^- \geq 0, (x^+)^T x^- = 0\}$$

where

$$\mathcal{S}_1 = \{ \varphi_1 \in (Mx^+ + Nx^- \leq b) \mid \mathbf{P}' \models \varphi_1 \},$$

$$\mathcal{S}_2 = \left\{ \varphi_2 \in (M'x^+ + N'x^- \leq b') \mid \begin{array}{l} \exists \varphi_1 \in (Mx^+ + Nx^- \leq b), \\ \gamma(\mathbf{P}) = \gamma((\mathbf{P} \setminus \{\varphi_1\}) \cup \{\varphi_2\}) \end{array} \right\}$$

Implementation and Experiments

Prototype

Prototype implementation **rAVI** using:

- GMP (the GNU Multiple Precision arithmetic library)
 - to guarantee the soundness of the implementation
- NewPolka: a rational implementation of the polyhedra domain
 - for Chernikova's algorithm

Interface:

- plugged into the APRON library [Jeannet Miné]
- programs analyzed with INTERPROC [Jeannet et al.]

Comparison with

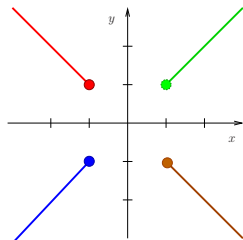
- NewPolka [Jeannet]
- itvPol: floating-point implementation of interval polyhedra [Chen et al. SAS09]

Example analyses

```

real x, y;
assume x = 1 or x = -1;
assume y = 1 or y = -1;
while (true) {
  ① if (x ≥ 0) { x := x + 1; }
    else      { x := x - 1; }
    if (y ≥ 0) { y := y + 1; }
    else      { y := y - 1; }
}

```



Loc	NewPolka	<i>itvPol</i>	rAVI
①	\top (no information)	$[-1, 1]x \leq -1$ $\wedge [-1, 1]y \leq -1$	$ x = y \wedge x \geq 1$

Preliminary experimental results

Program		NewPolka		itvPol		rAVI		Res. Inv.	
name	#vars	#iter.	$t(ms)$	#iter.	$t(ms)$	#iter.	$t(ms)$		
AVtest1	2	4	11	4	45	4	48	<	<
AVtest2	2	4	8	3	14	4	31	<	<
AVtest3	2	4	9	4	16	5	73	<	<
CmplxTest1	5	4	7	4	26	4	57	<	<
CmplxTest2	5	6	10	6	34	6	150	<	<
CmplxTest3	8	4	17	4	242	4	310	<	<
program4	1	5	2	4	4	4	10	<	=
program5	2	6	9	5	20	8	45	<	<

Most linear AV invariants captured by rAVI are essentially due to **piecewise linear** behaviors in the program, e.g., branches inside loops, case by case discussions over the difference between loop counter and input parameter (or initial value).

Conclusion

Summary:

- **goal:** handle **piecewise linear** behaviors in programs (non-convex)
- **approach:** **linear absolute value relation analysis**
 - show equivalence among itv linear, linear AV, extended LCP systems
 - develop a double description method for extended LCP
 - propose a new abstract domain: the AVI abstract domain

$$(\sum_k a_k x_k + \sum_k b_k |x_k| \leq c)$$

- can express **non-convex (even unconnected)** properties
- generalize the classical polyhedra abstract domain

Conclusion

Future Work

- **for precision**

- automatic methods to introduce auxiliary variables on the fly that can be used inside the AV function

- **for efficiency**

- weakly relational abstract domains over absolute value, with less expressiveness but higher efficiency
- floating-point implementation

- **new applications**

- program analysis of AV-related mathematical library functions
 - *abs*, *fdim*, *fmax*, *fmin*
- piece-wise linear abstraction for floating-point arithmetic
- analysis and verification of piece-wise linear (hybrid) systems