Principles of Abstract Interpretation

MIT press

Ch. 40, Zone and Octagon Analysis

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides-40--zone-octagon-analysis-PrAbsInt.pdf

Chapter 40

Ch. 40, Zone and Octagon Analysis (1/4)

We have split our review of chapter 40 into four videos

This first video is about

- zone properties

Zones and octagons

Zones

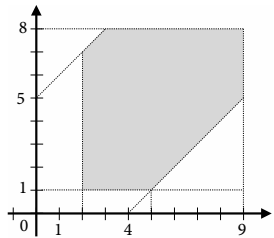
- Inequalities between values of variables are necessary e.g. to bound indexes used in the traversal of arrays with a symbolic bound n, as in

```
l1: {T} i = 0;
while l2: (i < n) {i>=0}
   l3: {i>=0, i<=n-1} i = (i + 1);
l4: {i>=0}
```

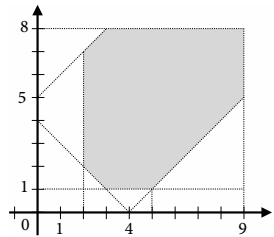
- Zones can express inequalities of the form $\pm x_i \leqslant c$ or $x_i - x_j \leqslant c$ where x_i and x_j are values of scalar variables $\mathtt{x}_i$, $\mathtt{x}_j$ and c is a scalar constant determined by the analysis.
- The canonical encoding of zones by normalized difference bounded matrices with an extra zero variable was introduced by Vaughan Pratt [Pratt, 1977].
- Zones were first used to analyze timed systems [Berthomieu and Menasche, 1983; Dill, 1989]. Octagons add constraints of the form $x_i + x_j \leqslant c$.

Octagons

- Octagons were first used for program parallelization in [Balasundaram and Kennedy, 1989] (where they are called “simple sections”).
- The missing operations to form an abstract domain, in particular assignment, test, join, widening, and narrowing were designed by Antoine Miné [Miné, 2001a,b, 2006].



Zone



Octagon

Zone analysis

Zone abstract domain

We study the zone abstract domain

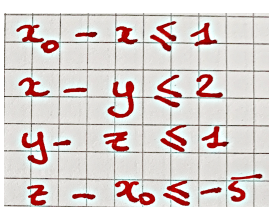
$$\mathbb{D}^{▱} \quad \triangleq \quad \langle \mathbb{P}^{▱}, \sqsubseteq^{▱}, \bot^{▱}, \sqcup^{▱}, \text{assign}_{▱}[\![x, A]\!], \text{test}^{▱}[\![B]\!], \overline{\text{test}}^{▱}[\![B]\!], \nabla^{▱}, \Delta^{▱} \rangle.$$

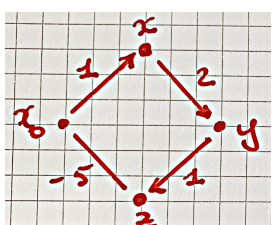
Zone abstract properties, section 40.1.1

- As in section 38.1, we let $\langle x_1, \ldots, x_m \rangle$ denote the values of the program variables $\mathcal{V} = \{\mathrm{x}_1, \ldots, \mathrm{x}_m\}$.
- More precisely, we consider the isomorphism between environment properties $P \in \mathbb{P} = \wp(\mathcal{V} \to \mathbb{F})$ and $\overline{P} \in \wp(\mathbb{F}^m)$ where
  - $\langle \mathbb{F}, \leqslant, +, -, \times, / \rangle$ is a totally ordered field,
  - $m = |\mathcal{V}|$ is the cardinality of the finite set $\mathcal{V}$ of variables and
  - $\rho \in P$ if and only if $\langle \rho(\mathrm{x}_1), \ldots, \rho(\mathrm{x}_m) \rangle \in \overline{P}$.
- The field $\mathbb{F}$ can be the integers $\mathbb{Z}$, the rationals $\mathbb{Q}$, or the reals $\mathbb{R}$.
- To express $\pm x_i \leqslant c$, we consider an additional variable x_0 which value is assumed to be always $0 \in \mathbb{F}$ so $x_i \leqslant c$ is $x_i - x_0 \leqslant c$ and $-x_i \leqslant c$ is $x_0 - x_i \leqslant c$.
- In this way all constraints are of the form $x_i - x_j \leqslant c_{ij}, i, j \in [0, m]$ and $i \neq j$.

- The constraints are represented by a weighted graph $G = \langle [0, m],\ E,\ \boldsymbol{\omega} \rangle$ (section 39.7) where
  - the set of vertices $[0, m]$ is isomorphic to the set of program variables $\{\mathtt{x}_0, \mathtt{x}_1, \ldots, \mathtt{x}_m\}$ augmented by $\mathtt{x}_0$ and
  - there is an edge $\langle i,\ j \rangle$ with weight $\boldsymbol{\omega}(\langle i,\ j \rangle) = c_{ij}$ for each constraint $x_i - x_j \leqslant c_{ij}$ between the values x_i, x_j of the program variables $\mathtt{x}_i$, $\mathtt{x}_j$, $i, j \in [0, m]$ and $i \neq j$.
- We let $\boldsymbol{\omega}(\langle i,\ i \rangle) = 0$, $i \in [0, m]$.
- The concretization of $G = \langle [0, m],\ E,\ \boldsymbol{\omega} \rangle$ is

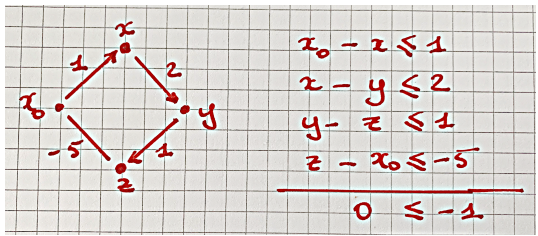
$$\gamma^{\square}(G) \quad \triangleq \quad \{\langle x_1,\ \ldots,\ x_m \rangle \mid \forall \langle i,\ j \rangle \in E\ .\ x_i - x_j \leqslant \boldsymbol{\omega}(\langle i,\ j \rangle) \wedge x_0 = 0\}$$



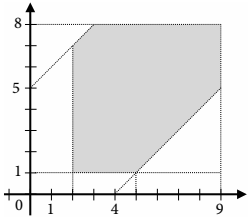


- If the graph G has a cycle $x_{i_1}, x_{i_2}, x_{i_3}, \ldots, x_{i_n}, x_{i_1}$ with strictly negative weight then $x_{i_1} - x_{i_2} \leqslant c_{i_1 i_2}$, $x_{i_2} - x_{i_3} \leqslant c_{i_2 i_3}$, …, $x_{i_n} - x_{i_1} \leqslant c_{i_n i_1}$ with $c_{i_1 i_2} + c_{i_2 i_3} + \ldots + c_{i_n i_1} < 0$ so that $0 = x_{i_1} - x_{i_2} + x_{i_2} - x_{i_3} + \ldots, + x_{i_n} - x_{i_1} \leqslant c_{i_1 i_2} + c_{i_2 i_3} + \ldots + c_{i_n i_1} < 0$, which is false, and therefore $\gamma^{\square}(G) = \varnothing$.
- Typically, weighted graphs are represented by their isomorphic adjacency-matrix (section 39.18) $\mathbf{G} = (\!| \langle i,\ j \rangle \in E \,?\, \boldsymbol{\omega}(i,j) : \infty |\!)_{\substack{i=1,m \\ j=1,m}} \in (\mathbb{F} \cup \{\infty\})^{n \times n}$ so that

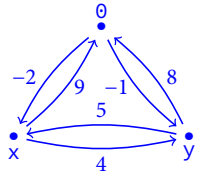
$$\gamma^{\square}(\mathbf{G}) \triangleq \{\langle x_1,\ \ldots,\ x_m \rangle \mid \forall i, j \in [0,m]\ .\ x_i - x_j \leqslant \mathbf{G}_{ij} \wedge x_0 = 0\}.$$



Example




```
x>=2, y>=1,
x<=9, x-y<=4,
y<=8, y-x<=5
```



$$\begin{array}{c} \\ 0 \\ x \\ y \end{array} \begin{array}{c} \begin{array}{ccc} 0 & x & y \end{array} \\ \begin{bmatrix} 0 & -2 & -1 \\ 9 & 0 & 4 \\ 8 & 5 & 0 \end{bmatrix} \end{array}$$

$$\begin{aligned} 0 - x &\leqslant -2 \quad & 0 - y &\leqslant -1, \\ x - 0 &\leqslant 9 \quad & x - y &\leqslant 4, \\ y - 0 &\leqslant 8 \quad & y - x &\leqslant 5 \end{aligned}$$

Normalization

- Even if the graph G has no cycle with strictly negative weight, several different graphs may have the same concretization.
- For example $x - y \leqslant 0 \wedge y - z \leqslant 1$ and $x - y \leqslant 0 \wedge y - z \leqslant 1 \wedge x - z \leqslant 1$.
- So we need to normalize the abstract domain as in section 36.3.3.
- For that purpose all graphs with a cycle of strictly negative weight are normalized to the infimum $\bot^{\square}$.
- All other graphs G with no cycle of strictly negative weight (represented by their adjacency matrix $\mathbf{G}$ such that $\gamma^{\square}(G) = \gamma^{\square}(\mathbf{G})$) are normalized to a unique matrix $\mathrm{norm}^{\square}(\mathbf{G})$ which is
  - meaning-preserving i.e. $\gamma^{\square}(\mathbf{G}) = \gamma^{\square}(\mathrm{norm}^{\square}(\mathbf{G}))$, and
  - normalizing i.e. $\gamma^{\square}(\mathrm{norm}^{\square}(\mathbf{G}_1)) = \gamma^{\square}(\mathrm{norm}^{\square}(\mathbf{G}_2))$ implies $\mathrm{norm}^{\square}(\mathbf{G}_1) = \mathrm{norm}^{\square}(\mathbf{G}_2)$ (and conversely).

  (It follows that $\mathrm{norm}^{\square}$ is idempotent).

- The set of normalized zone abstract properties is

$$\overline{\mathbb{P}}^{\square} \triangleq \{\bot^{\square}\} \cup \{\mathbf{D} \in (\mathbb{F} \cup \{\infty\})^{m+1\times m+1} \mid \forall i \in [0,m] \,.\, \mathbf{D}_{ii} = 0 \wedge \mathrm{norm}^{\square}(\mathbf{D}) = \mathbf{D}\}.$$

with

$$\begin{array}{rcl} \gamma^{\square}(\mathbf{D}) & \triangleq & \{\langle x_1, \ldots, x_m\rangle \mid \forall i, j \in [0,m] \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0\} \\ \gamma^{\square}(\bot^{\square}) & \triangleq & \varnothing \end{array}$$

where, for all $v \in \mathbb{F}$, $v \leqslant \infty$ and $\infty \leqslant \infty$ are always true.

- The normalization by saturation adds as much true constraints as possible by computing the distance matrix with Roy-Floyd-Warshall algorithm (theorem 39.26)
- the normalization by minimization (of [Larsen, Pettersson, and W. Yi, 1995] suggested by [Bagnara, Hill, Mazzi, and Zaffanella, 2005]) looks for a minimal number of constraints.

$$\begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & \infty & 0 \end{bmatrix}$$

`{x=1, y=2, x<=y-1}`

Distance graph

$$\begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}$$

`{x=1, y=2, x=y-1}`

Normalization by saturation

$$\begin{bmatrix} 0 & \infty & -2 \\ 1 & 0 & \infty \\ \infty & 1 & 0 \end{bmatrix}$$

`{y>=2, x<=1, y<=x+1}`

Normalization by minimization

- We prefer the normalization by saturation which introduces redundant constraints that may be propagated by incomplete assignment and test transformers and that would be definitely missed with the normalization by minimization.

Normalization by saturation

- In the normalization by saturation, the graph G (as represented by its adjacency matrix $\mathbf{G}$) is normalized to its distance graph $G^* = \langle [0,m],\ [0,m] \times [0,m],\ \mathrm{d} \rangle$ (as represented by its matrix of distances[1] (section 39.18) $\mathbf{D} \in (\mathbb{F} \cup \{\infty\})^{m+1 \times m+1}$).
- The normal form is computed by the Roy-Floyd-Warshall algorithm $\mathbf{D} = \mathsf{RFW}(\mathbf{G})$ such that

$$\begin{array}{rcll} \mathsf{RFW}(\mathbf{D}) & \triangleq & \bot^{\square} & \text{if } \mathbf{D} \text{ has a cycle of strictly negative weight} \\ \mathsf{RFW}(\mathbf{D})_{ij} & \triangleq & \min(\mathbf{D}_{ij}, \min\limits_{k \in [0,m]} (\mathbf{D}_{ik} + \mathbf{D}_{kj})) & \text{otherwise, (corollary 39.40)} \end{array}$$

where, for all $v \in \mathbb{F}$, $v \leqslant \infty$ and $\infty \leqslant \infty$ are always true.

Lemma (40.2, meaning preservation) $\forall \mathbf{D} \in (\mathbb{F} \cup \{\infty\})^{m+1 \times m+1} \,.\, \gamma^{\square}(\mathsf{RFW}(\mathbf{D})) = \gamma^{\square}(\mathbf{D})$.

[1] called the difference bounded matrix in [Dill, 1989].

Proof of lemma 40.2 – RFW($\mathbf{D}$) returns $\bot^{\square}$ if and only if the zone constrains are not satisfiable so the concretization is $\varnothing$ in both cases.

– Otherwise, there is no cycle with negative weight

– So if the graph G has a cycle $x_{i_1}, x_{i_2}, x_{i_3}, \ldots, x_{i_n}, x_{i_1}$ then $x_{i_1} - x_{i_2} + x_{i_2} - x_{i_3} + \ldots, + x_{i_n} - x_{i_1} \leqslant c_{i_1 i_2} + c_{i_2 i_3} + \ldots + c_{i_n i_1} \geqslant 0$ and therefore setting $\text{RFW}(\mathbf{D})_{i_1 i_1} = 0$ does not change the concretization since $x_{i_1} - x_{i_1} = 0$.

– So before the main loop on k of the Roy-Floyd-Warshall algorithm $\gamma^{\square}(\mathbf{D}) = \gamma^{\square}(\mathbf{D}^0)$ where $\mathbf{D}^0$ is the initial matrix.

– Let us show that a loop iteration does not change this invariant and so holds upon termination of the for loops.

– Assume by induction hypothesis that $\gamma^{\square}(\mathbf{D}) = \gamma^{\square}(\mathbf{D}^0)$.

– Then $\mathbf{D}$ is changed into $\mathbf{D}'_{ij} = \min(\mathbf{D}_{ij}, (\mathbf{D}_{ik} + \mathbf{D}_{kj}))$

– so $\gamma^{\square}(\mathbf{D}') = \gamma^{\square}(\mathbf{D}) = \gamma^{\square}(\mathbf{D}^0)$ since if $\mathbf{D}'_{ij} \neq \mathbf{D}_{ij}$ then $\mathbf{D}'_{ij} = \mathbf{D}_{ik} + \mathbf{D}_{kj}$ so by def. $\gamma^{\square}$ the term $x_i - x_j \leqslant \mathbf{D}'_{ij}$ in the concretization of $\mathbf{D}'$ is logically equivalent to the term $x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_i - x_k \leqslant \mathbf{D}_{ik} \wedge x_k - x_j \leqslant \mathbf{D}_{kj} \Leftrightarrow x_i - x_k \leqslant \mathbf{D}_{ik} \wedge x_k - x_j \leqslant \mathbf{D}_{kj}$ appearing in the concretization of $\mathbf{D}$. □

Lemma (40.3, normalization) $\gamma^{\square}(\text{RFW}(\mathbf{G}_1)) = \gamma^{\square}(\text{RFW}(\mathbf{G}_2)) \Leftrightarrow \text{RFW}(\mathbf{G}_1) = \text{RFW}(\mathbf{G}_2)$.

Proof of lemma 40.3 – For the non-trivial implication $\Rightarrow$, there must exist i, j such that $\text{RFW}(\mathbf{G}_1)_{ij} \neq \text{RFW}(\mathbf{G}_2)_{ij}$, say $\text{RFW}(\mathbf{G}_1)_{ij} < \text{RFW}(\mathbf{G}_2)_{ij}$, since $\mathbb{F} \cup \{\infty\}$ is totally ordered.

– Since the concretizations are the same, the graphs are the same,

– So by theorem 39.33, the shortest distance between i and j should be the same in both cases, a contradiction. □

Lattice of zones

- Since an operation on normalized zones may return a non-normalized zone, the results on all operations on zones must be normalized, if necessary.
- The zone union $\sqcup^{\diamond}$ is an example of operation which returns a normalized zone for normalized parameters.
- On the contrary, the intersection $\sqcap^{\diamond}$ of normalized zones may not be normalized, which requires a renormalization.
- However, this normalization may prevent the convergence of the widening, as shown and solved in section $40.1.8$.

Theorem (40.4, lattice of zones) The domain of zone properties $\langle \mathbb{P}^{\square}, \sqsubseteq^{\square}, \bot^{\square}, \top^{\square}, \sqcup^{\square}, \sqcap^{\square} \rangle$ and the subdomain of normalized zone properties $\langle \overline{\mathbb{P}}^{\square}, \sqsubseteq^{\square}, \bot^{\square}, \top^{\square}, \sqcup^{\square}, \sqcap^{\square} \rangle$ are lattices where, for all $\mathbf{D}, \mathbf{D}' \in \mathbb{P}^{\square}$,

$$
\begin{array}{rcl}
\bot^{\square} \sqsubseteq^{\square} \mathbf{D} \sqsubseteq^{\square} \top^{\square} & & \\
\mathbf{D} \sqsubseteq^{\square} \mathbf{D}' & = & \forall i, j \in [0, m] \,.\, \mathbf{D}_{ij} \leqslant \mathbf{D}'_{ij} \\
\mathbf{D} = \mathbf{D}' & \triangleq & \forall i, j \in [0, m] \,.\, \mathbf{D}_{ij} = \mathbf{D}'_{ij} \\
\top^{\square} & \triangleq & (\infty)^{\,m+1 \times m+1} \\
\mathbf{D} \sqcup^{\square} \mathbf{D}' & \triangleq & (\max(\mathbf{D}_{ij}, \mathbf{D}'_{ij}))^{\,m+1 \times m+1} \\
\mathbf{D} \sqcap^{\square} \mathbf{D}' & \triangleq & (\min(\mathbf{D}_{ij}, \mathbf{D}'_{ij}))^{\,m+1 \times m+1} \in \mathbb{P}^{\square} \\
& = & \mathsf{RFW}\left((\min(\mathbf{D}_{ij}, \mathbf{D}'_{ij}))^{\,m+1 \times m+1}\right) \in \overline{\mathbb{P}}^{\square}
\end{array}
\tag{40.4}
$$

Proof of (40.4) — Partial order: We define $P \sqsubseteq^{\square} Q$ if and only if $\gamma^{\square}(P) \subseteq \gamma^{\square}(Q)$. We have $\gamma^{\square}(\bot^{\square}) = \varnothing$ and $\gamma^{\square}(\top^{\square}) = \mathbb{F}^m$ so $\bot^{\square} \sqsubseteq^{\square} \mathbf{D} \sqsubseteq^{\square} \top^{\square}$ since $\gamma^{\square}(\mathbf{D}) \in \wp(\mathbb{F}^m)$. Moreover

$$
\begin{array}{ll}
& \mathbf{D} \sqsubseteq^{\square} \mathbf{D}' \\
\triangleq & \gamma^{\square}(\mathbf{D}) \subseteq \gamma^{\square}(\mathbf{D}') \qquad \text{(def. } \sqsubseteq^{\square}\text{)} \\
\Leftrightarrow & (\forall i, j \in [0, m] \,.\, x_i - x_j \leqslant \mathbf{D}_{ij}) \Rightarrow (\forall i, j \in [0, m] \,.\, x_i - x_j \leqslant \mathbf{D}'_{ij}) \\
& \qquad \text{(def. } \gamma^{\square} \text{ and } \subseteq\text{)} \\
\Leftrightarrow & \forall i, j \in [0, m] \,.\, \mathbf{D}_{ij} \leqslant \mathbf{D}'_{ij} \qquad \text{(def. } \leqslant\text{)}
\end{array}
$$

— Equality: $P = Q$ is $P \sqsubseteq^{\square} Q \wedge Q \sqsubseteq^{\square} P$ and follows from the unicity of the matrix of distances returned by Roy-Floyd-Warshall algorithm.

— Least upper bound:

– We have $\mathbf{D}_{ij} \leqslant \max(\mathbf{D}_{ij}, \mathbf{D}'_{ij})$ and $\mathbf{D}'_{ij} \leqslant \max(\mathbf{D}_{ij}, \mathbf{D}'_{ij})$ so that, by def. $\gamma^{\square}$, $\mathbf{D} \sqsubseteq^{\square} \mathbf{D} \sqcup^{\square} \mathbf{D}'$ and $\mathbf{D}' \sqsubseteq^{\square} \mathbf{D} \sqcup^{\square} \mathbf{D}'$.

– Given an upper bound $\mathbf{D}''$ of $\mathbf{D}$ and $\mathbf{D}'$, we have $\mathbf{D}_{ij} \leqslant \mathbf{D}''_{ij}$ and $\mathbf{D}'_{ij} \leqslant \mathbf{D}''_{ij}$ so that $\max(\mathbf{D}_{ij}, \mathbf{D}'_{ij}) \leqslant \mathbf{D}''_{ij}$ proving that $\mathbf{D} \sqcup^{\square} \mathbf{D}' \sqsubseteq^{\square} \mathbf{D}''$, and therefore that $\mathbf{D} \sqcup^{\square} \mathbf{D}'$ is the least upper bound.

– Let $\mathbf{D}'' = \mathbf{D} \sqcup^{\square} \mathbf{D}'$. We must prove that $\mathbf{D}'' \in \overline{\mathbb{P}}^{\square}$.

– Obviously $\mathbf{D}''_{ii} = \max(\mathbf{D}_{ii}, \mathbf{D}'_{ii}) = \max(0, 0) = 0$.

– We must show that $\mathrm{RFW}(\mathbf{D}'') = \mathbf{D}''$.

– If not, we have some $\mathbf{D}''_{ij} \neq \min(\mathbf{D}''_{ij}, \min_{k\in[0,m]}(\mathbf{D}''_{ik} + \mathbf{D}''_{kj}))$ so $\min_{k\in[0,m]}(\mathbf{D}''_{ik} + \mathbf{D}''_{kj}) < \mathbf{D}''_{ij}$, hence $\mathbf{D}''_{ik} + \mathbf{D}''_{kj} < \mathbf{D}''_{ij}$ for some $k \in [0, m]$.

– By def. of $\sqcup^{\square}$, this implies that $\max(\mathbf{D}_{ik}, \mathbf{D}'_{ik}) + \max(\mathbf{D}_{kj}, \mathbf{D}'_{kj}) < \max(\mathbf{D}_{ij}, \mathbf{D}'_{ij})$.

– If $\max(\mathbf{D}_{ij}, \mathbf{D}'_{ij}) = \mathbf{D}_{ij}$ then $\mathbf{D}_{ik} + \mathbf{D}_{kj} < \mathbf{D}_{ij}$, in contradiction with $\mathrm{RFW}(\mathbf{D}) = \mathbf{D}$.

– Otherwise, $\max(\mathbf{D}_{ij}, \mathbf{D}'_{ij}) = \mathbf{D}'_{ij}$ and then $\mathbf{D}'_{ik} + \mathbf{D}'_{kj} < \mathbf{D}'_{ij}$, in contradiction with $\mathrm{RFW}(\mathbf{D}') = \mathbf{D}'$.

– By contradiction, $\mathrm{RFW}(\mathbf{D}'') = \mathbf{D}''$.

— Greatest lower bound:

– By $\leqslant$-order duality, $\mathbf{D}'' = \min(\mathbf{D}_{ij}, \mathbf{D}'_{ij}))^{m+1\times m+1}$ is a $\sqsubseteq^{\square}$-glb of $\mathbf{D}$ and $\mathbf{D}'$ with 0-diagonal.

– However, it might not be in normal form.

– An example is $x - y \leqslant 1$ and $y - z \leqslant 2$ for which the meet yields

$x - y \leqslant 1 \wedge y - z \leqslant 2 \wedge x - z \leqslant \min(\infty, \infty) = \infty$ and not $x - z \leqslant 3$.

– So the meet is $\mathsf{RFW}(\mathbf{D}'') \in \mathbb{P}^{\square}$ such that $\gamma^{\square}(\mathbf{D}'') = \gamma^{\square}(\mathsf{RFW}(\mathbf{D}''))$ by the normalization lemma 40.2. □

This concludes our definition of

- zone properties

from chapter 40, "Zone and Octagon Analysis"

The End

Principles of Abstract Interpretation

MIT press

Ch. 40, Zone and Octagon Analysis

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides-40--zone-octagon-analysis-PrAbsInt.pdf

Chapter 40

Ch. 40, Zone and Octagon Analysis (2/4)

In this second video, we study

- zone transformers

Calculational design of the zone abstract assignment transformers

- Following (21.7), we look for $\mathrm{assign}^{\square}_{\square}[\![x, A]\!]$ satisfying $\mathrm{assign}^{\square}_{r}[\![x, A]\!] \circ \gamma^{\square} \subseteq \gamma^{\square} \circ \mathrm{assign}^{\square}_{\square}[\![x, A]\!]$, by calculational design.
- The assertional assignment transformer $\mathrm{assign}^{\square}_{r}[\![x, A]\!]$ in (19.12) is $\varnothing$-strict, so we get

$$\begin{aligned} \mathrm{assign}^{\square}_{\square}[\![x_k, A]\!]\, \bot^{\square} &\triangleq \bot^{\square} \\ \mathrm{test}^{\square}[\![B]\!]\, \bot^{\square} &\triangleq \bot^{\square} \\ \overline{\mathrm{test}}^{\square}[\![B]\!]\, \bot^{\square} &\triangleq \bot^{\square}. \end{aligned}$$

For the invertible assignment ($c \in \mathbb{F}$ is the value denoted by c and $\mathbf{D} \neq \bot^{\square}$),

$$(\text{assign}_{\square}[\![\mathrm{x}_k, \mathrm{x}_k + \mathrm{c}]\!]\ \mathbf{D})_{ij} \triangleq \begin{cases} \mathbf{D}_{ij} + c & \text{if } i = k \wedge j \neq k \\ \mathbf{D}_{ij} - c & \text{if } i \neq k \wedge j = k \\ \mathbf{D}_{ij} & \text{otherwise} \end{cases} \tag{40.7}$$

we have $\text{assign}_{\vec{r}}[\![\mathrm{x}_k, \mathrm{x}_k + \mathrm{c}]\!] \circ \gamma^{\square} = \gamma^{\square} \circ \text{assign}_{\square}[\![\mathrm{x}_k, \mathrm{x}_k + \mathrm{c}]\!]$.

Proof sketch of (40.7)

— $x'_k - x_j \leqslant \mathbf{D}_{kj}$ ⟅precondition before the assignment⟆

⇒ $(x_k - c) - x_j \leqslant \mathbf{D}_{kj}$

⟅postcondition after the assignment so $x_k = x'_k + c$ so $x'_k = x_k - c$⟆

⇒ $x_k - x_j \leqslant \mathbf{D}_{kj} + c$

— $x_i - x'_k \leqslant \mathbf{D}_{ik}$ ⟅precondition before the assignment⟆

⇒ $x_i - (x_k - c) \leqslant \mathbf{D}_{ik}$

⟅postcondition after the assignment so $x_k = x'_k + c$ so $x'_k = x_k - c$⟆

⇒ $x_i - x_k + c \leqslant \mathbf{D}_{ik}$

⇒ $x_i - x_k \leqslant \mathbf{D}_{ik} - c$

Observe that if $\mathbf{D} \in \overline{\mathbb{P}}^{\square}$ then $\text{assign}_{\square}[\![\mathsf{x}_k, \mathsf{x}_k + \mathsf{c}]\!]\ \mathbf{D} \in \overline{\mathbb{P}}^{\square}$ so no normalization is necessary. □

For the elimination of a variable x_k, define

$$(\mathsf{elim}^{\square}\ \mathsf{x}_k\ \mathbf{D})_{ij} \triangleq \begin{cases} 0 & \text{if } i = k \text{ and } j = k \\ \infty & \text{else, if } i = k \text{ or } j = k \\ \mathbf{D}_{ij} & \text{otherwise} \end{cases} \tag{40.8}$$

such that

$$\mathsf{elim}^{\vec{r}}\ \mathsf{x} \circ \gamma^{\square} = \gamma^{\square} \circ \mathsf{elim}^{\square}\ \mathsf{x}$$

where, up to the encoding of variable values in section 40.1.1,

$$\mathsf{elim}^{\vec{r}}\ \mathsf{x}_k\ P \triangleq \{\langle \mathsf{x}_1, \ \ldots, \ \mathsf{x}_k, \ \ldots, \ \mathsf{x}_m \rangle \mid \exists v \in \mathbb{F} \ . \ \langle \mathsf{x}_1, \ \ldots, \ \mathsf{x}_{k-1}, \ v, \ \mathsf{x}_{k+1}, \ \ldots, \ \mathsf{x}_m \rangle \in P\}.$$

Proof of (40.8)

$$\mathsf{elim}^{\vec{\mathsf{r}}}\ \mathsf{x}_k\ (\gamma^{\varpi}(\mathbf{D}))$$

$$= \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \exists x_k \in \mathbb{F} \,.\, \langle x_1, \ldots, x_k, \ldots, x_m\rangle \in \gamma^{\varpi}(\mathbf{D})\}$$

(def. $\mathsf{elim}^{\vec{\mathsf{r}}}$)

$$= \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \exists x_k \in \mathbb{F} \,.\, \forall i, j \in [0, m] \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0\}$$

(def. γ^{ϖ})

$$= \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \forall i, j \in [0, m] \setminus \{k\} \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0\}$$

($\mathbf{D} \neq \bot^{\varpi}$)

$$= \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \forall i, j \in [0, m] \setminus \{k\} \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0 \wedge v - v = 0 \wedge \forall i \in [0, m] \setminus \{k\} \,.\, x_i - v \leqslant \infty \wedge \forall j \in [0, m] \setminus \{k\} \,.\, v - x_j \leqslant \infty\}$$

(def. 0 and ∞)

$\{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \forall i, j \in [0,m] \setminus \{k\} \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0 \wedge v - v = 0 \wedge \forall i \in [0,m] \setminus \{k\} \,.\, x_i - v \leqslant \infty \wedge \forall j \in [0,m] \setminus \{k\} \,.\, v - x_j \leqslant \infty\}$

$= \{\langle x_1, \ldots, x_k, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \forall i, j \in [0,m] \setminus \{k\} \,.\, x_i - x_j \leqslant \mathbf{D}_{ij} \wedge x_0 = 0 \wedge x_k - x_k = 0 \wedge \forall i \in [0,m] \setminus \{k\} \,.\, x_i - x_k \leqslant \infty \wedge \forall j \in [0,m] \setminus \{k\} \,.\, x_k - x_j \leqslant \infty\}$ ⟅letting $v = x_k$⟆

$= \{\langle x_1, \ldots, x_k, \ldots, x_m\rangle \mid \forall i, j \in [0,m] \,.\, x_i - x_j \leqslant \mathsf{elim}^{\square}\ \mathsf{x}_k\ \mathbf{D}\}$ ⟅def. (40.8) of $\mathsf{elim}^{\square}$⟆

$= \gamma^{\square}(\mathsf{elim}^{\square}\ \mathsf{x}_k\ \mathbf{D})$ ⟅def. $\gamma^{\square}$⟆

$\mathsf{elim}^{\square}\ \mathsf{x}_k\ \mathbf{D} \in \overline{\mathbb{P}}^{\square}$ since it satisfies the normal form condition of corollary 39.40 when $\mathbf{D} \in \overline{\mathbb{P}}^{\square}$ does. □

For the non-invertible assignment $\ell \neq k$,

$$(\text{assign}_{\mathbin{\square}}[\![x_k, x_\ell + c]\!]\ \mathbf{D})_{ij} \triangleq \text{let } \mathbf{D}' = \text{elim}^{\mathbin{\square}}(x_k)\ \mathbf{D} \text{ in } \begin{cases} \mathbf{D}'_{i\ell} - c & \text{if } i \neq k \text{ and } j = k \\ \mathbf{D}'_{\ell j} + c & \text{if } i = k \text{ and } j \neq k \\ \mathbf{D}'_{ij} & \text{otherwise} \end{cases} \tag{40.9}$$

Informal proof of (40.9)

Proof Sketch

- x_k may have been modified so all contraints in $\mathbf{D}$ relating another variable to x_k may no longer be valid and so are eliminated;
- So $\mathbf{D}'$ holds after the assignment as well as $x_k = x_\ell + c$ (and so $x_\ell = x_k - c$);
- So all variables related to x_ℓ are also related to x_k and normalization with saturation will introduce such relations;
- However, it is cheaper to directly compute them as follows:
  - $x_i - x_\ell \leqslant \mathbf{D}'_{i\ell}$
    $\Rightarrow x_i - (x_k - c) \leqslant \mathbf{D}'_{i\ell}$
    $\Rightarrow x_i - x_k \leqslant \mathbf{D}'_{i\ell} - c$ and so, $\mathbf{D}'_{ik}$ (which was ∞) becomes $\mathbf{D}'_{i\ell} - c$;
  - $x_\ell - x_j \leqslant \mathbf{D}'_{\ell j}$
    $\Rightarrow (x_k - c) - x_j \leqslant \mathbf{D}'_{\ell j}$
    $\Rightarrow x_k - x_j \leqslant \mathbf{D}'_{\ell j} + c$ and so, $\mathbf{D}'_{kj}$ (which was ∞) becomes $\mathbf{D}'_{\ell j} + c$;
  - No other constraint can be added to $\mathbf{D}'$ which is saturated . □

Proof of (40.9)

$$\mathsf{assign}_{\vec{r}}[\![\mathsf{x}_k, \mathsf{x}_\ell + \mathsf{c}]\!](\gamma^{\square}(\mathbf{D})) \qquad \wr k \neq \ell \wr$$

$$= \{\langle x_1, \ldots, x_{k-1}, x_\ell + c, x_{k+1}, \ldots, x_m\rangle \mid \exists v \,.\, \langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \in \gamma^{\square}(\mathbf{D})\}$$

$\wr$ def. (19.12) of $\mathsf{assign}_{\vec{r}}[\![\mathsf{x}, \mathsf{A}]\!]$, def. (3.4) of $\mathscr{A}$ and $\mathscr{A}[\![\mathsf{c}]\!]\rho = c$, encoding of variable values in section 40.1.1, and letting v to be the value of variable x_k before the assignment $\wr$

$$= \{\langle x_1, \ldots, x_{k-1}, x_\ell + c, x_{k+1}, \ldots, x_m\rangle \mid \exists v \,.\, \langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \in \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \exists x_k \in \mathbb{F} \,.\, \langle x_1, \ldots, x_k, \ldots, x_m\rangle \in \gamma^{\square}(\mathbf{D})\}\}$$

$\wr$ def. $\in$ and letting $v = x_k$ $\wr$

$$= \{\langle x_1, \ldots, x_{k-1}, x_\ell + c, x_{k+1}, \ldots, x_m\rangle \mid \langle x_1, \ldots, x_m\rangle \in \mathsf{elim}^{\vec{r}}(\mathsf{x}_k)(\gamma^{\square}(\mathbf{D}))\}$$

$\wr$ def. (40.8) of $\mathsf{elim}^{\vec{r}}$ $\wr$

$$= \{\langle x_1, \ldots, x_{k-1}, x_\ell + c, x_{k+1}, \ldots, x_m\rangle \mid \langle x_1, \ldots, x_\ell, \ldots, x_m\rangle \in \gamma^{\square}(\mathsf{elim}^{\square}(\mathsf{x}_k)(\mathbf{D}))\}$$

$\wr$ def. (40.8) of $\mathsf{elim}^{\square}$ $\wr$

$$\{\langle x_1, \ldots, x_{k-1}, x_\ell + c, x_{k+1}, \ldots, x_m\rangle \mid \langle x_1, \ldots, x_\ell, \ldots, x_m\rangle \in \gamma^{\square}(\mathrm{elim}^{\square}(\mathsf{x}_k)\,(\mathbf{D}))\}$$

$$= \{\langle x'_1, \ldots, x'_{k-1}, x'_\ell, x'_{k+1}, \ldots, x'_m\rangle \mid \langle x'_1, \ldots, x'_{\ell-1}, x'_\ell - c, x'_{\ell+1}, \ldots, x'_m\rangle \in \gamma^{\square}(\mathrm{elim}^{\square}(\mathsf{x}_k)\,(\mathbf{D}))\}$$

$$\text{letting } x'_i = x_i \text{ for } i \neq \ell \text{ and } x'_\ell = x_\ell + c$$

$$= \{\langle x_1, \ldots, x_\ell, \ldots, x_m\rangle \mid \forall i, j \in [0,m] \setminus \{\ell\} \,.\, x_i - x_j \leqslant \mathbf{D}'_{ij} \wedge \forall i \in [0,m] \,.\, x_i - (x_\ell - c) \leqslant \mathbf{D}'_{i\ell} \wedge \forall j \in [0,m] \,.\, (x_\ell - c) - x_j \leqslant \mathbf{D}'_{\ell j} \wedge x_0 = 0\}$$

$$\text{def. } \gamma^{\square} \text{ where } \mathbf{D}' = \mathrm{elim}^{\square}(\mathsf{x}_k)\,(\mathbf{D})$$

$$= \{\langle x_1, \ldots, x_\ell, \ldots, x_m\rangle \mid \forall i, j \in [0,m] \setminus \{\ell\} \,.\, x_i - x_j \leqslant \mathbf{D}'_{ij} \wedge \forall i \in [0,m] \,.\, x_i - x_\ell \leqslant \mathbf{D}'_{i\ell} - c \wedge \forall j \in [0,m] \,.\, x_\ell - x_j \leqslant \mathbf{D}'_{\ell j} + c \wedge x_0 = 0\}$$

$$\text{def. + and } - \text{ in totally ordered field } \mathbb{F}$$

$$= \{\langle x_1, \ldots, x_m\rangle \mid \forall i, j \in [0,m] \,.\, x_i - x_j \leqslant (\mathrm{assign}_{\square}[\![\mathsf{x}_k, \mathsf{x}_\ell + \mathsf{c}]\!]\,\mathbf{D}')_{ij} \wedge x_0 = 0\}$$

$$\text{by defining } (\mathrm{assign}_{\square}[\![\mathsf{x}_k, \mathsf{x}_\ell + \mathsf{c}]\!]\,\mathbf{D}')_{ij} \text{ as in (40.7)}$$

$$= \gamma^{\square}(\mathrm{assign}_{\square}[\![\mathsf{x}_k, \mathsf{x}_k + \mathsf{c}]\!]\,\mathbf{D}') \qquad \text{def. } \gamma^{\square} \quad \square$$

- As a special case $\ell = 0$ such that $x_0 = 0$, we get

$$(\text{assign}_{\square}[\![\mathsf{x}_k, \mathsf{c}]\!]\ \mathbf{D})_{ij} \triangleq \text{let } \mathbf{D}' = \text{elim}^{\square}(\mathsf{x}_k)\ \mathbf{D} \text{ in } \begin{cases} -c & \text{if } i \neq 0 \text{ and } j = 0 \\ +c & \text{if } i = 0 \text{ and } j \neq 0 \\ \mathbf{D}'_{ij} & \text{otherwise} \end{cases}$$

- Otherwise,

$$\text{assign}_{\square}[\![\mathsf{x}_k, \mathsf{A}]\!]\ \mathbf{D} \triangleq \text{elim}^{\square}\ \mathsf{x}_k\ \mathbf{D} \tag{40.10}$$

Remark 1 Following section 38.4.3, non-linear assignments may be linearizable in one of the form (40.7) or (40.9), in particular when replacing constant variables by their value, as possibly determined by the zone analysis. □

Proof of (40.10)

$$
\begin{aligned}
& \mathsf{assign}_{\vec{r}}[\![\mathsf{x}_k, \mathsf{A}]\!](\gamma^{\square}(\mathbf{D})) \\
= \; & \{\rho[\mathsf{x}_k \leftarrow \mathscr{A}[\![\mathsf{A}]\!]\rho] \mid \rho \in \gamma^{\square}(\mathbf{D})\} && \text{⟅def. (19.12) of } \mathsf{assign}_{\vec{r}}[\![\mathsf{x}, \mathsf{A}]\!]\text{⟆} \\
= \; & \{\langle x_1, \ldots, x_{k-1}, \mathscr{A}[\![\mathsf{A}]\!]\langle x_1, \ldots, x_m\rangle, x_{k+1}, \ldots, x_m\rangle \mid \langle x_1, \ldots, x_k, \ldots, x_m\rangle \in \gamma^{\square}(\mathbf{D})\} \\
& && \text{⟅encoding of variable values in section } 40.1.1\text{⟆} \\
\subseteq \; & \{\langle x_1, \ldots, x_{k-1}, v, x_{k+1}, \ldots, x_m\rangle \mid v \in \mathbb{F} \wedge \langle x_1, \ldots, x_k, \ldots, x_m\rangle \in \gamma^{\square}(\mathbf{D})\} \\
& && \text{⟅def. } \subseteq \text{ and } \mathscr{A}[\![\mathsf{A}]\!]\langle x_1, \ldots, x_m\rangle \in \mathbb{F}\text{⟆} \\
= \; & \mathsf{elim}^{\vec{r}}\, \mathsf{x}_k\, (\gamma^{\square}(\mathbf{D})) && \text{⟅def. (40.8) of } \mathsf{elim}^{\vec{r}}\text{⟆} \\
= \; & \gamma^{\square}(\mathsf{elim}^{\square}\, \mathsf{x}_k\, \mathbf{D}) && \text{⟅def. (40.8) of } \mathsf{elim}^{\square}\text{⟆} \\
= \; & \gamma^{\square}(\mathsf{assign}_{\square}[\![\mathsf{x}_k, \mathsf{A}]\!]\, \mathbf{D}) && \text{⟅def. (40.10) of } \mathsf{assign}_{\square}[\![\mathsf{x}_k, \mathsf{A}]\!]\text{⟆} \quad \square
\end{aligned}
$$

Calculational design of the zone abstract test transformers

- For the test (>= is similar and = reduces to two inequalities),

$$(\text{test}^{\square}[\![\mathsf{x}_k - \mathsf{x}_\ell <= \mathsf{c}]\!]\mathbf{D})_{ij} \triangleq \begin{cases} \min(\mathbf{D}_{ij}, c) & \text{if } i = k \text{ and } j = \ell \\ \mathbf{D}_{ij} & \text{otherwise} \end{cases}$$

$$(\text{test}^{\square}[\![\mathsf{x}_k <= \mathsf{c}]\!]\mathbf{D})_{ij} \triangleq \begin{cases} \min(\mathbf{D}_{ij}, c) & \text{if } i = k \text{ and } j = 0 \\ \mathbf{D}_{ij} & \text{otherwise} \end{cases}$$

which is $\bot^{\square}$ if this creates a cycle with strictly negative weight. Otherwise,

$$\text{test}^{\square}[\![\mathsf{B}]\!]\mathbf{D} \triangleq \mathbf{D}$$
$$\overline{\text{test}}^{\square}[\![\mathsf{B}]\!]\mathbf{D} \triangleq \mathbf{D}$$

This concludes our definition of

- zone transformers

from chapter 40, "Zone and Octagon Analysis"

The End

Principles of Abstract Interpretation

MIT press

Ch. 40, Zone and Octagon Analysis

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides-40--zone-octagon-analysis-PrAbsInt.pdf

Chapter 40

Ch. 40, Zone and Octagon Analysis (3/4)

In this third video, we study

- zone widening and narrowing

Zone widening and narrowing, section 40.1.8

The widening and narrowing are inspired by those of intervals in chapter 32 where unstable constraints are eliminated.

$$\bot^{\square} \nabla^{\square} \mathbf{D}' \triangleq \mathbf{D}'$$

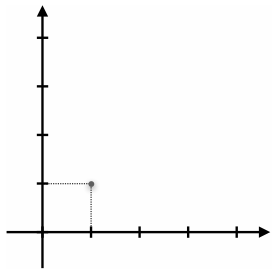
$$\mathbf{D} \nabla^{\square} \bot^{\square} \triangleq \mathbf{D}$$

$$(\mathbf{D} \nabla^{\square} \mathbf{D}')_{ij} \triangleq \begin{cases} \mathbf{D}_{ij} & \text{if } \mathbf{D}'_{ij} \leqslant \mathbf{D}_{ij} \\ \infty & \text{otherwise} \end{cases}$$

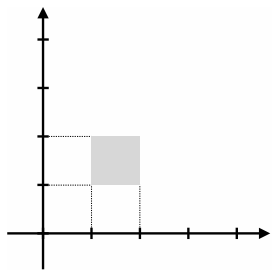
$$(\mathbf{D} \Delta^{\square} \mathbf{D}')_{ij} \triangleq \begin{cases} \mathbf{D}'_{ij} & \text{if } \mathbf{D}_{ij} = \infty \\ \mathbf{D}_{ij} & \text{otherwise} \end{cases}$$

Example

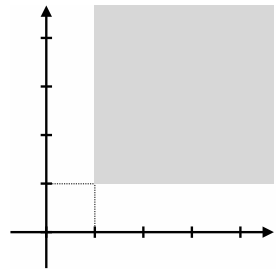
The widening on zones is similar to the widening on intervals.



$\nabla^{\square}$



=



$$\begin{pmatrix} 0 & -1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

{y>=1, x<=1, y<=x}

$\nabla^{\square}$

$$\begin{pmatrix} 0 & -1 & -1 \\ 2 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix}$$

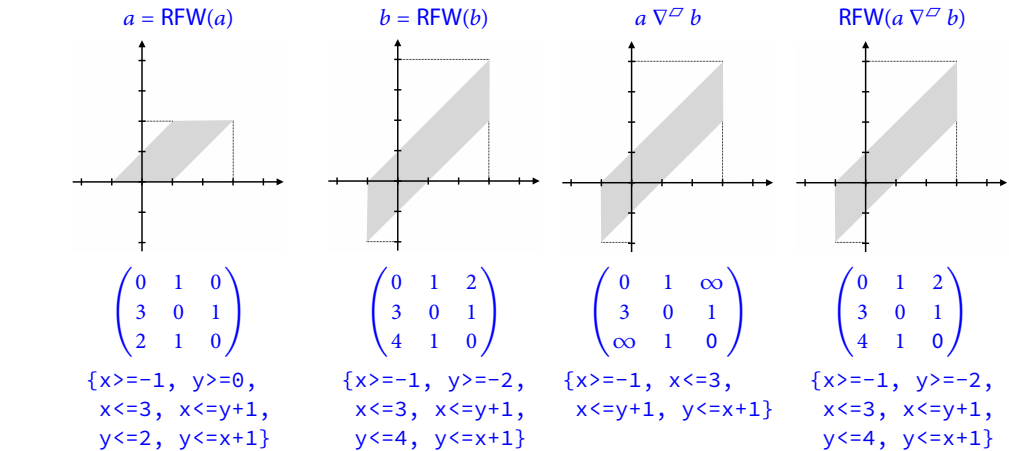
{x>=1, y>=1, x<=2, y<=2}

=

$$\begin{pmatrix} 0 & -1 & -1 \\ \infty & 0 & \infty \\ \infty & \infty & 0 \end{pmatrix}$$

{x>=1, y>=1}

- Normalization by saturation may prevent convergence of the iteration sequence with widening



- The problem is that normalization by saturation reintroduces the constraints on y eliminated by the widening to ∞.
- A solution is to limit normalizations [Miné, 2006].
- It is also possible to use normalization by minimization of constraints [Bagnara, Hill, Mazzi, and Zaffanella, 2005] but this potentially impairs the precision of incomplete transformers.
- Another solution is to use an history widening of section 34.7, where the constraints eliminated by the previous widening are recorded and cannot be reintroduction by the normalization.

- The upward iteration with history widening of section 34.7 of $\overline{f} \in \overline{L} \to \overline{L}$ from $a \in \overline{L}$ with widening ∇ and normalization norm on a poset $\langle \overline{L}, \overline{\sqsubseteq} \rangle$ is modified as follows $\langle \hat{f}^n, n \in \mathbb{N}_\omega \rangle$ with an abstraction of the iteration history as follows:

(1) $\hat{f}^0 \triangleq \text{norm}(a)$,

(2) $\hat{f}^{n+1} \triangleq (\!| \overline{f}(\hat{f}^n) \mathrel{\overline{\sqsubseteq}} \hat{f}^n ?\; \hat{f}^n$ when $n \in \mathbb{N}$,
$\quad : \text{norm}(\hat{f}^n \nabla \overline{f}(\hat{f}^n)) \sqcup \bigsqcup_{0 \leqslant \delta < n} \hat{f}^\delta \nabla \overline{f}(\hat{f}^\delta) \,|\!)$

(3) $\hat{f}^\omega \triangleq \text{norm}(\bigtriangledown_{n<\omega} \hat{f}^n)$.

Theorem (40.14) The zone history iterates with widening $\nabla^{\square}$ and normalization RFW do converge in finitely many steps (so that case 3 is not used).

Proof of theorem 40.14 – If the iteration is not stabilized, the widening extrapolate at least one element of the matrix to ∞.

– Even if it is eliminated by the normalization, joining with the term $\bigsqcup_{0 \leqslant \delta < n} \hat{f}^{\delta} \nabla \overline{f}(\hat{f}^{\delta})$ ensures that from the next iterate on, the component will stay stabilized to ∞.

– Because the matrix is finite of dimension $(m+1)^2$, this ensures convergence in at most $(m+1)^2$ steps. □

This concludes our definition of

- zone widening and narrowing

from chapter 40, "Zone and Octagon Analysis"

The End

Principles of Abstract Interpretation

MIT press

Ch. 40, Zone and Octagon Analysis

Patrick Cousot

pcousot.github.io

PrAbsInt@gmail.com github.com/PrAbsInt/

These slides are available at
http://github.com/PrAbsInt/slides/slides-40--zone-octagon-analysis-PrAbsInt.pdf

Chapter 40

Ch. 40, Zone and Octagon Analysis (4/4)

In this fourth video, we study

- octagon analysis

Octagon Analysis

- Octagons include constraints of the form $x_i + x_j \leqslant c$ where x_i and x_j are values of variables $\mathtt{x}_i$ and $\mathtt{x}_j$ and $c \in \mathbb{V} \cup \{\infty\}$ is a constant inferred by the analysis.
- The encoding proposed by Antoine Miné [Miné, 2004, Ch. 4.2] consists in using only constraints of the form $\pm x - \pm y \leqslant c$ where x and y are values of variables and $c \in \mathbb{V}$ is a constant.
- The translation is the following

$$\begin{array}{rcl@{\qquad}rcl} x \leqslant c & \rightarrow & x - (-x) \leqslant 2c & x \geqslant c & \rightarrow & (-x) - x \leqslant -2c \\ x - y \leqslant c & \rightarrow & x - y \leqslant c & x - y \geqslant c & \rightarrow & y - x \leqslant -c \\ x + y \leqslant c & \rightarrow & x - (-y) \leqslant c & x + y \geqslant c & \rightarrow & (-x) - y \leqslant -c \end{array}$$

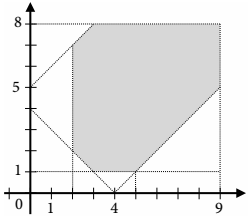
- This transformation results is distance constraints $x - y \leqslant c$ between values x_i of variables $\mathtt{x}_i$ and their opposite value $-x_i$.
- This can be encoded by a distance matrix $\mathbf{M} \in (\mathbb{V} \cup \{\infty\})^{2m}$ where m is the number of variables where x_i is at line/column position i and x_i is at line/column position $m + i$.

- The concretization is

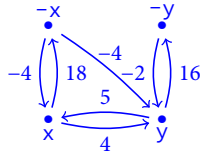
$$\gamma^{\hexagon}(\mathbf{M}) \triangleq \{\langle x_1, \ldots, x_m\rangle \mid \forall i, j \in [1, 2m] \,.\, v(x,i) - v(x,j) \leqslant \mathbf{M}_{ij}\}$$

  where $v(x,i) = \llbracket\, i \leqslant m \,?\, x_i : -x_{i-m} \,\rrbracket$.
- Two different components of the matrix matrix may express the same constraint $x - y \leqslant c \Leftrightarrow (-y) - (-x) \leqslant c$.
- This is taken into account by requiring that $\forall i, j \in [1, m] \,.\, \mathbf{M}_{ij} = \mathbf{M}_{m+j,m+i}$.

Example




```
x>=2, y>=1,
x<=9, x-y<=4,
y<=8, y-x<=5,
x+y>=4
```



$$\begin{array}{c} \\ x \\ y \\ -x \\ -y \end{array} \begin{array}{c} \begin{array}{cccc} x & y & -x & -y \end{array} \\ \begin{bmatrix} 0 & 4 & -18 & \infty \\ 5 & 0 & \infty & 16 \\ -4 & -4 & 0 & 5 \\ \infty & 16 & 4 & 0 \end{bmatrix} \end{array}$$

- The lattice structure is similar to that for zones.
- The normalization by saturation is performed by a variant of the Roy-Floyd-Warshall algorithm [Miné, 2006].
- The assignment and test transformers are also similar to the zone case [Miné, 2006].
- The normalization after widening of unstable elements to ∞ may impair convergence so a history widening must be used.

Conclusion

Conclusion I

- More details on zones and octagons are provided in [Chawdhary, Robbins, and King, 2014; Miné, 2001a, 2006], proofs details are found in [Miné, 2004].
- [Miné, 2001a, 2004] use normalization by saturation while [Bagnara, Hill, Mazzi, and Zaffanella, 2005] suggested normalization by minimization.
- [Miné, 2004, Definition 3.6.4] proposes an abstraction of linear assignments more precise that (40.10).
- An implementation of octagons is certified in [Jourdan, 2016; Jourdan, Laporte, Blazy, Leroy, and Pichardie, 2015].
- Sparse [Gange, Navas, Schachte, Søndergaard, and Stuckey, 2016; Jourdan, 2017] and compact [Chawdhary and King, 2017] representations have been proposed for the zone distance matrix.
- All result remain valid when replacing $\leqslant$ by strict equalities $<$, but not both.
- [Péron and Halbwachs, 2007] adds disequality constraints.

Conclusion II

- Vector instructions may be used for fast implementations of the matrix operations [Banterle and Giacobazzi, 2007].
- Zone and octagon abstract domains have many applications such as resource and complexity analysis [Sinn, Zuleger, and Veith, 2017].
- Zone/octagon and more generally relational analyses of program with thousands of variables hardly scale up.
- It is therefore necessary to look for relations not between all variables but between some sets of variables (called "packs" or "variable-clusters") of the program.
- For large programs, these packs may be different at different program points.
- The packs can be determined statically (before the analysis) e.g. based on where relations between variables might be useful (e.g. between array indexes and the symbolic bounds of the array) [P. Cousot, R. Cousot, Feret, Mauborgne, Miné, and Rival, 2009; Heo, Oh, and K. Yi, 2017].

Conclusion III

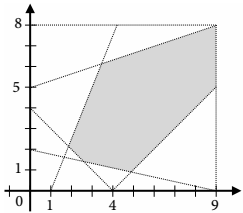
- The packs may also be determined dynamically during the analysis, as proposed by [Heo, Oh, and Yang, 2016].
- For a given cluster of variables, [Singh, Püschel, and Vechev, 2015, 2018] consider a further decomposition of the relational domains into independent components thus speeding up the analysis.
- All these ideas rely on a meta-analysis (i.e. a static/offline or dynamic/omline analysis of the iterative static analysis) as shown by [**DBLP:journals/pacmpl/CousotGR19**].
- Depending on the required precision of the linear relational analysis, several other abstract domains can be used beyond intervals, zones, octagons, and linear equalities such as pentagons [Logozzo and Fähndrich, 2010], parallelotopes [Amato, Rubino, and Scozzari, 2017], octahedra [Clarisó and Cortadella, 2007], two variables per inequality [Simon and King, 2010], gauge [Venet, 2012], etc., see [Miné, 2017] for a tutorial on inferring numerical properties of programs.

Conclusion IV

- They all consist in considering specific templates of polyhedra [P. Cousot and Halbwachs, 1978].
- Generalizing predicate abstraction of section 27.7, the templates may be inferred by a pre-analysis or specified by the end-user e.g. [Sankaranarayanan and Sassi, 2017; Sankaranarayanan, Sipma, and Manna, 2004].
- Unfortunately, the exactly appropriate templates can only be found from the program proof, the chicken or the egg causality dilemma.

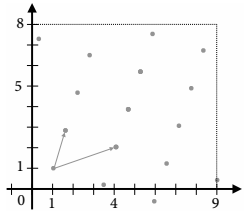
Conclusion V

- There are many other linear or non-linear relational domains:



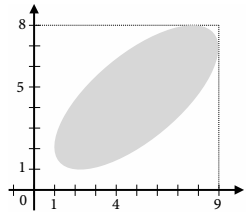
Polyhedra

[P. Cousot and Halbwachs, 1978]



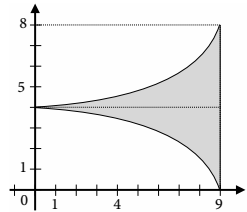
Linear congruences

[Granger, 1991]



Ellipses

[Feret, 2004]



Exponentials

[Feret, 2005]

References I

Bibliography

Amato, Gianluca, Marco Rubino, and Francesca Scozzari (2017). "Inferring Linear Invariants with Parallelotopes.". *Sci.Comput.Program.* 148, pp. 161–188.

Bagnara, Roberto, Patricia M. Hill, Elena Mazzi, and Enea Zaffanella (2005). "Widening Operators for Weakly–Relational Numeric Abstractions.". In *SAS*. Vol. 3672. Lecture Notes in Computer Science. Springer, pp. 3–18.

Balasundaram, Vasanth and Ken Kennedy (1989). "A Technique for Summarizing Data Access and Its Use in Parallelism Enhancing Transformations.". In *PLDI*. ACM, pp. 41–53.

Banterle, Francesco and Roberto Giacobazzi (2007). "A Fast Implementation of the Octagon Abstract Domain on Graphics Hardware.". In *SAS*. Vol. 4634. Lecture Notes in Computer Science. Springer, pp. 315–332.

References II

Berthomieu, Bernard and Miguel Menasche (1983). "An Enumerative Approach for Analyzing Time Petri Nets.". In *IFIP Congress*. Pp. 41–46.

Chawdhary, Aziem and Andy King (2017). "Compact Difference Bound Matrices.". In *APLAS*. Vol. 10695. Lecture Notes in Computer Science. Springer, pp. 471–490.

Chawdhary, Aziem, Edward Robbins, and Andy King (2014). "Simple and Efficient Algorithms for Octagons.". In *APLAS*. Vol. 8858. Lecture Notes in Computer Science. Springer, pp. 296–313.

Clarisó, Robert and Jordi Cortadella (2007). "The Octahedron Abstract Domain.". *Sci.Comput.Program.* 64.1, pp. 115–139.

Cousot, Patrick, Radhia Cousot, Jérôme Feret, Laurent Mauborgne, Antoine Miné, and Xavier Rival (2009). "Why Does Astrée Scale Up?.". *Formal Methods in System Design*. 35.3, pp. 229–264.

Cousot, Patrick and Nicolas Halbwachs (1978). "Automatic Discovery of Linear Restraints Among Variables of a Program.". In *POPL*. ACM Press, pp. 84–96.

References III

Dill, David L. (1989). "Timing Assumptions and Verification of Finite–State Concurrent Systems.". In *Automatic Verification Methods for Finite State Systems*. Vol. 407. Lecture Notes in Computer Science. Springer, pp. 197–212.

Feret, Jérôme (2004). "Static Analysis of Digital Filters.". In *ESOP*. Vol. 2986. Lecture Notes in Computer Science. Springer, pp. 33–48.

— (2005). "The Arithmetic–Geometric Progression Abstract Domain.". In *VMCAI*. Vol. 3385. Lecture Notes in Computer Science. Springer, pp. 42–58.

Gange, Graeme, Jorge A. Navas, Peter Schachte, Harald Søndergaard, and Peter J. Stuckey (2016). "Exploiting Sparsity in Difference–Bound Matrices.". In *SAS*. Vol. 9837. Lecture Notes in Computer Science. Springer, pp. 189–211.

Granger, Philippe (1991). "Static Analysis of Linear Congruence Equalities Among Variables of a Program.". In *TAPSOFT, Vol.1*. Vol. 493. Lecture Notes in Computer Science. Springer, pp. 169–192.

References IV

Heo, Kihong, Hakjoo Oh, and Hongseok Yang (2016). "Learning a Variable–Clustering Strategy for Octagon From Labeled Data Generated by a Static Analysis.". In *SAS*. Vol. 9837. Lecture Notes in Computer Science. Springer, pp. 237–256.

Heo, Kihong, Hakjoo Oh, and Kwangkeun Yi (2017). "Selective Conjunction of Context–Sensitivity and Octagon Domain Toward Scalable and Precise Global Static Analysis.". *Softw., Pract.Exper.* 47.11, pp. 1677–1705.

Jourdan, Jacques–Henri (May 26, 2016). "Verasco: A Formally Verified C Static Analyzer.(Verasco: un analyseur statique pour C formellement vérifié).". PhD thesis. Université Paris Diderot.

— (2017). "Sparsity Preserving Algorithms for Octagons.". *Electr.Notes Theor.Comput.Sci.* 331, pp. 57–70.

Jourdan, Jacques–Henri, Vincent Laporte, Sandrine Blazy, Xavier Leroy, and David Pichardie (2015). "A Formally–Verified C Static Analyzer.". In *POPL*. ACM, pp. 247–259.

Larsen, Kim Guldstrand, Paul Pettersson, and Wang Yi (1995). "Compositional and Symbolic Model Checking of Real–Time Systems.". In *RTSS*. IEEE Computer Society, pp. 76–87.

References V

Logozzo, Francesco and Manuel Fähndrich (2010). "Pentagons: A Weakly Relational Abstract Domain for the Efficient Validation of Array Accesses.". *Sci.Comput.Program.* 75.9, pp. 796–807.

Miné, Antoine (2001a). "A New Numerical Abstract Domain Based on Difference–Bound Matrices.". In *PADO*. Vol. 2053. Lecture Notes in Computer Science. Springer, pp. 155–172.

— (2001b). "The Octagon Abstract Domain.". In *Proceedings of the Eighth Working Conference on Reverse Engineering, WCRE'01, Stuttgart, Germany, October 2-5, 2001*. IEEE Computer Society, pp. 310–319.

— (Dec. 6, 2004). "Weakly Relational Numerical Abstract Domains.". Thèse de doctorat. Palaiseau, France: École polytechnique.

— (2006). "The Octagon Abstract Domain.". *Higher–Order and Symbolic Computation.* 19.1, pp. 31–100.

— (2017). "Tutorial on Static Inference of Numeric Invariants by Abstract Interpretation.". *Foundations and Trends in Programming Languages.* 4.3–4, pp. 120–372.

References VI

Péron, Mathias and Nicolas Halbwachs (2007). "An Abstract Domain Extending Difference–Bound Matrices with Disequality Constraints.". In *VMCAI*. Vol. 4349. Lecture Notes in Computer Science. Springer, pp. 268–282.

Pratt, Vaughan R. (Sept. 1977). "Two Easy Theories Whose Combination Is Hard.". MIT. `http://boole.stanford.edu/pub/sefnp.pdf`.

Sankaranarayanan, Sriram and Mohamed Amin Ben Sassi (2017). "Template Polyhedra with a Twist.". In *SAS*. Vol. 10422. Lecture Notes in Computer Science. Springer, pp. 321–341.

Sankaranarayanan, Sriram, Henny B. Sipma, and Zohar Manna (2004). "Constraint–Based Linear–Relations Analysis.". In *SAS*. Vol. 3148. Lecture Notes in Computer Science. Springer, pp. 53–68.

Simon, Axel and Andy King (2010). "The Two Variable Per Inequality Abstract Domain.". *Higher–Order and Symbolic Computation*. 23.1, pp. 87–143.

Singh, Gagandeep, Markus Püschel, and Martin T. Vechev (2015). "Making Numerical Program Analysis Fast.". In *PLDI*. ACM, pp. 303–313.

References VII

Singh, Gagandeep, Markus Püschel, and Martin T. Vechev (2018). "A Practical Construction for Decomposing Numerical Abstract Domains.". *Proc.ACM Program.Lang.* 2.POPL, 55:1–55:28.

Sinn, Moritz, Florian Zuleger, and Helmut Veith (2017). "Complexity and Resource Bound Analysis of Imperative Programs Using Difference Constraints.". *J.Autom.Reasoning.* 59.1, pp. 3–45.

Venet, Arnaud J. (2012). "The Gauge Domain: Scalable Analysis of Linear Inequality Invariants.". In *CAV*. Vol. 7358. Lecture Notes in Computer Science. Springer, pp. 139–154.

Home work

Read Ch. 40 “Zone and Octagon Analysis” of

Principles of Abstract Interpretation
Patrick Cousot
MIT Press

The End, Thank you