A SIMULATION TEST BETWEEN ZONES

Problem: Given two zones Z and Z', we want to check if every region that intersects Z, also intersects Z'.

The goal of these notes is to provide an algorithm for the above problem, which runs in time $O(|X|^2)$ where X is the set of clocks.

- As seen in a previous lecture, this test can be used in the reachability algorithm to ensure correctness and termination of the zone enumeration.

- A preliminary version of this test appears in the following paper:

  Using non-convex approximations for efficient analysis of timed automata
  – Herbreteau, Kini, Srivathsan, Walukiewicz
  FSTTCS '11

- The test has been polished and extended to several settings since then.

Plan:

- 1. Some definitions, and the actual test
- 2. Illustration of the test on some examples
- 3. Proof of correctness

Part 1: Some definitions and the actual test

Fix a set of clocks X

Bounds function: A bounds function $M : X \rightarrow \mathbb{N}$ associates a natural number to each clock.

For convenience, we will write M_x for $M(x)$, where $x \in X$.

Region equivalence: Given a bounds function M. We say

$$v \simeq_M v' \quad \text{if}$$

1. $\forall x \in X$: $\quad v(x) \leq M_x \quad$ iff $\quad v'(x) \leq M_x$

2. $\forall x \in X$ s.t. $v(x) \leq M_x$:

$$\lfloor v(x) \rfloor = \lfloor v'(x) \rfloor$$

$$\{v(x)\} = 0 \quad \text{iff} \quad \{v'(x)\} = 0$$

3. $\forall x, y \in X$ s.t. $v(x) \leq M_x$ and $v(y) \leq M_y$:

$$\{v(x)\} \leq \{v(y)\} \quad \text{iff} \quad \{v'(x)\} \leq \{v'(y)\}$$

We will call the equivalence classes of $\simeq_M$ as **M-regions**.

Sometimes, we will simply write regions when M is clear from the context.

Exercise: Let $v \simeq_M v'$, and $x, y \in X$ s.t. $v(x) \leq M_x$, $v(y) \leq M_y$

Show that: (i) $\{v(x)\} < \{v(y)\} \iff \{v'(x)\} < \{v'(y)\}$

(ii) $\{v(x)\} = \{v(y)\} \iff \{v'(x)\} = \{v'(y)\}$

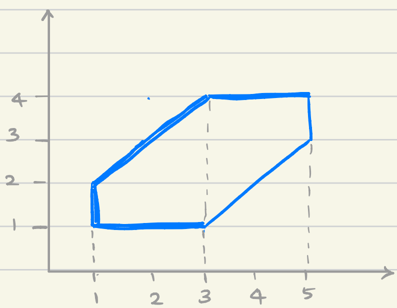
Representing zones:

Recall that zones are sets of valuations represented using conjunctions of constraints of the form:

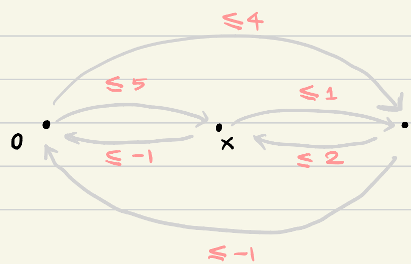
$x \sim c$ and $x - y \sim c$. where $\sim \in \{<, \leq, >, \geq\}$

In this document we will assume $c \in \mathbb{Z}$

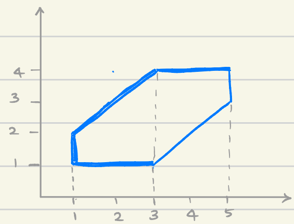
We will represent zones using distance graphs. Here is an example.



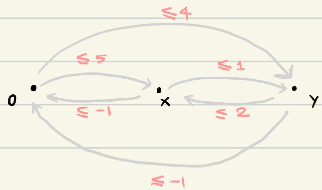
$$\begin{aligned} x &\geq 1 \quad \wedge \\ y - x &\leq 1 \quad \wedge \\ y &\leq 4 \quad \wedge \\ x &\leq 5 \quad \wedge \\ x - y &\leq 2 \quad \wedge \\ y &\geq 1 \end{aligned}$$



Distance graphs:



$$x \geq 1 \;\wedge\; y - x \leq 1 \;\wedge\; y \leq 4 \;\wedge\; x \leq 5 \;\wedge\; x - y \leq 2 \;\wedge\; y \geq 1$$



A distance graph has vertices $\{0\} \cup X$.

Edges are directed and carry a weight of the form:

$$(\triangleleft, c) \;\cup\; \{(<, \infty)\}$$

where $c \in \mathbb{Z}$ and $\triangleleft \in \{<, \leq\}$

$$\boxed{x \xrightarrow{\triangleleft\, c} y \quad \text{represents} \quad y - x \triangleleft c}$$

Above example gives a zone and its distance graph.

For a distance graph G, we define: $[\![G]\!] = \{ v \mid v \text{ satisfies } y - x \triangleleft c$ for every $x \xrightarrow{\triangleleft c} y$ in $G \}$

Arithmetic on weights:

We want to be able to manipulate conjunctions of constraints using distance graphs.

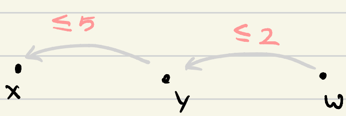
For example: $x - y \leq 5 \;\wedge$
$y - w \leq 2$

implies $x - w \leq 7$

whereas: $x - y \leq 5 \;\wedge$
$y - w < 2$

implies $x - w < 7$

At the level of graphs, if we have:



We should derive an edge $w \longrightarrow x$ with weight ≤ 7.

- This first calls for the definition of an addition over these weights.

Let $c, c_1, c_2 \in \mathbb{Z}$, $\triangleleft, \triangleleft_1, \triangleleft_2 \in \{<, \leq\}$

$$(\triangleleft, c) + (<, \infty) = (<, \infty)$$

$$(\triangleleft_1, c_1) + (\triangleleft_2, c_2) = \begin{cases} (<, c_1 + c_2) & \text{if either } \triangleleft_1 \text{ or } \triangleleft_2 \text{ is } < \\ (\leq, c_1 + c_2) & \text{otherwise} \end{cases}$$

We also need a way to compare constraints.

For example:

- $x - y \leq 2$ implies $x - y \leq 4$
- $x - y \leq 2$ implies $x - y < 3$
- $x - y < 2$ implies $x - y \leq 2$

We will define an order among weights that reflects this implication.

Let $c, c_1, c_2 \in \mathbb{Z}$, $\triangleleft, \triangleleft_1, \triangleleft_2 \in \{<, \leq\}$

$$(\triangleleft, c) < (<, \infty)$$

$$(\triangleleft_1, c_1) < (\triangleleft_2, c_2) \quad \text{if} \quad c_1 < c_2 \quad \text{or}$$

$$c_1 = c_2 \text{ and } \triangleleft_1 = < \text{, } \triangleleft_2 = \leq$$

The total order on weights looks like this.

$$\cdots \; (\leq, -2) \; (<, -1) \; (\leq, -1) \; (<, 0) \; (\leq, 0) \; (<, 1) \; (\leq, 1) \; (<, 2) \; \cdots \; (<, \infty)$$

Negative cycles:

- A path in a distance graph is a sequence of edges.
- Weight of a path is the sum of weight of its edges.

For eg: $x \xrightarrow{<2} y \xrightarrow{\leq -1} w$ has weight $(<, 1)$

- A cycle is a path that starts and ends with the same vertex.

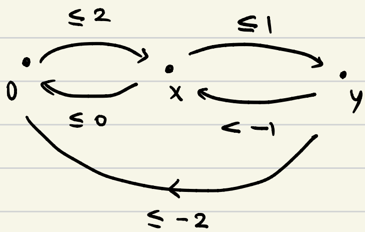
> A cycle in a distance graph is said to negative if its weight is less than or equal to $(<, 0)$

$$\begin{tikzcd} x \arrow[r, bend left, "{(\leq, 1)}"] & y \arrow[l, bend left, "{(\leq, -1)}"] \end{tikzcd} \qquad\qquad \begin{tikzcd} x \arrow[r, bend left, "{(\leq, 1)}"] & y \arrow[l, bend left, "{(<, -1)}"] \end{tikzcd}$$

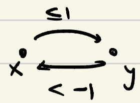
is NOT negative   is negative

- Negative cycles denote contradictions in the system of constraints.

For example:



has a negative cycle

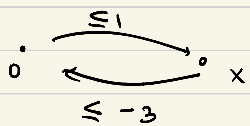


No valuation can satisfy

$$y - x \leq 1$$

and $$x - y < -1$$

Similarly:



is a negative cycle

representing:

$$x \leq 1$$
$$-x \leq -3 \qquad (x \geq 3)$$
$$\downarrow$$
a contradiction.

Here is a theorem that formalizes this observation.

Theorem: Let G be a distance graph.

$[\![G]\!]$ is non-empty iff all cycles in G are non-negative.

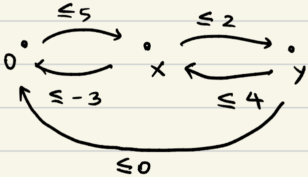
Intersection of distance graphs:

Let G_1, G_2 be distance graphs. Define

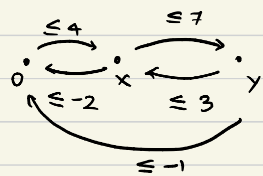
> $\min(G_1, G_2)$ to be the graph where weight of each edge is given by the minimum of the corresponding weights in G_1, G_2.

Eg:

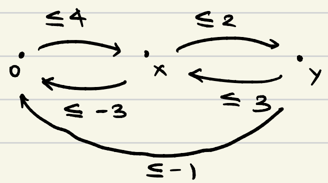
G_1



G_2



$\min(G_1, G_2)$



Note:

- When we do not draw an edge, the weight is assumed to be $(<, \infty)$. For eg. in the above graphs, weight of $0 \longrightarrow y$ is $(<, \infty)$

Min graph represents the intersection of the two sets.

Lemma: $[\![\min(G_1, G_2)]\!] = [\![G_1]\!] \cap [\![G_2]\!]$

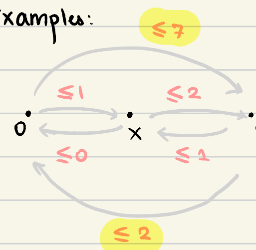
Canonical distance graphs:

A distance graph with no negative cycles is said to be in **canonical form** if

for every pair $x, y \in X \cup \{0\}$,
the shortest path from x to y is given by

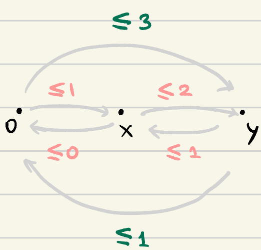
the edge: $x \longrightarrow y$.

Examples:



is not canonical due to the highlighted weights.

Canonical form of above graph is:



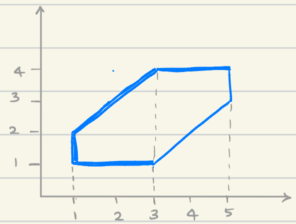
- Given a distance graph, its canonical form can be computed in $O(|X|^3)$ using Floyd-Warshall's all-pairs shortest path algo. This algorithm can also detect the presence of negative cycles.

Canonical distance graph of a zone:

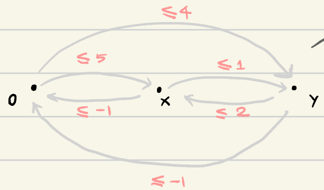
For a zone Z, we denote by G_Z its canonical distance graph.

> We write Z_{xy} for the weight of the $x \longrightarrow y$ edge in G_Z

For example: let Z be the zone given below:



$$Z = \left\{\begin{array}{l} x \geqslant 1 \;\wedge \\ y - x \leqslant 1 \;\wedge \\ y \leqslant 4 \;\wedge \\ x \leqslant 5 \;\wedge \\ x - y \leqslant 2 \;\wedge \\ y \geqslant 1 \end{array}\right.$$



G_Z

$Z_{0x} = (\leqslant, 5)$, $\qquad Z_{x0} = (\leqslant, -1)$

$Z_{xy} = (\leqslant, 1)$, $\qquad Z_{yx} = (\leqslant, 2)$

$Z_{0y} = (\leqslant, 4)$, $\qquad Z_{y0} = (\leqslant, -1)$

Region-closure inclusion: Given zones Z, Z', define:

$$\boxed{Z \sqsubseteq_M Z' \quad \text{if} \quad \forall v \in Z \quad \exists v' \in Z' \quad \text{s.t.} \quad v \simeq_M v'.}$$

From the definition, it is direct to see that $Z \sqsubseteq_M Z'$ iff for all M-regions R:

$$R \cap Z \neq \phi \quad \Rightarrow \quad R \cap Z' \neq \phi.$$

This give the following lemma:

Lemma: Let Z, Z' be non-empty zones.

$Z \not\sqsubseteq_M Z'$ iff $\exists$ an M-region R s.t.

$$R \cap Z \neq \phi \quad \text{and} \quad R \cap Z' = \phi$$

We will now state the main theorem:

Theorem: Let Z, Z' be non-empty zones.

$Z \not\sqsubseteq_M Z'$ iff $\exists\, x, y \in X \cup \{0\}$ s.t.

$$Z_{x0} + (\leq, M_x) \geq (\leq, 0) \quad \text{and}$$

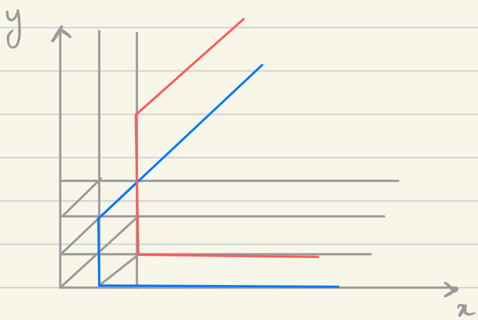
$$Z'_{xy} < Z_{xy} \quad \text{and}$$

$$Z'_{xy} + (<, -M_y) < (\leq, 0)$$

Part 2: Illustrating the test on some examples.

Example 1:

$M_x = 2, \qquad M_y = 3$



Blue zone: z

Red zone: z'

$z \not\subseteq_M z'$ due to the following witnesses:

- $z_{x0} + (\le, M_x) \succcurlyeq (\le, 0) \;\wedge\; z'_{x0} < z_{x0}$

$z_{x0} \downarrow (\le, -1)$, $\quad (\le, M_x) \downarrow (\le, 2)$, $\quad z'_{x0} \downarrow (\le, -2)$, $\quad z_{x0} \downarrow (\le, -1)$

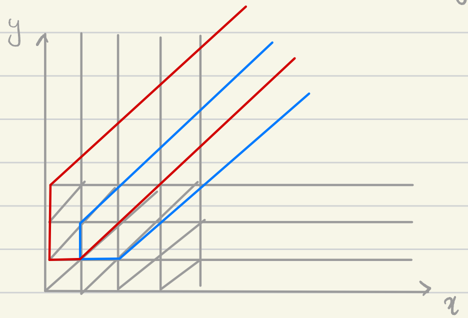
- $z_{y0} + (\le, M_y) \succcurlyeq (\le, 0) \;\wedge\; z'_{y0} < z_{y0}$

z_{y0}: $(\le, 0)$, $\quad (\le, M_y)$: $(\le, 3)$, $\quad z'_{y0}$: $(\le, -1)$, $\quad z_{y0}$: $(\le, 0)$

Exercise: Are there other (2-variable) witnesses?

Example 2:

$M_x = 4, \quad M_y = 3$



Blue : Z

Red : Z'

$Z \not\subseteq_M Z'$ because:

$$\underset{(\le,-1)}{Z_{y0}} + \underset{(\le,3)}{(\le, M_y)} \ge (\le, 0) \;\wedge\; \underset{(\le,0)}{Z'_{yx}} < \underset{(\le,1)}{Z_{yx}} \;\wedge\; \underset{(\le,0)}{Z'_{yx}} + \underset{(\le,-4)}{(<, -M_x)} < (\le, 0)$$

Exercise: Are there any other (2-variable) witnesses?