

Reachability Analysis for Timed Automata using Max-Plus Algebra

Qi Lu, Michael Madsen, Martin Milata, Søren Ravn,
Uli Fahrenberg, Kim G. Larsen

November 11, 2011

- 1 Timed automata
- 2 Max-plus algebra
- 3 Reachability in TA using max-plus polyhedra

Clock constraints $\mathcal{B}(X)$ generated by

$$g ::= x_1 \sim n \mid x_1 - x_2 \sim n \mid g_1 \wedge g_2$$

where $x_1, x_2 \in X$, $n \in \mathbb{N}$, $\sim \in \{\leq, =, \geq\}$.

A **timed automaton** is a quintuple (L, X, ℓ_0, E, I) , where

- L is a finite set of **locations**,
- X is a finite set of clocks,
- $\ell_0 \in L$ is the **initial location**,
- $E \subseteq L \times \mathcal{B}(X) \times 2^X \times L$ is the set of **edges** and
- $I : L \rightarrow \mathcal{B}(X)$ a function which assigns to every location an **invariant**.

Semantics in terms of transition system (S, s_0, \rightarrow) with:

- $S = \{(\ell, \nu) \mid \ell \in L, \nu : X \rightarrow \mathbb{R}_{\geq 0}, \nu \models I(\ell)\}$ are the **states**,
- $s_0 = (\ell_0, \nu_0)$, where $\nu_0(x) = 0$ for any x (**initial state**),
- $\rightarrow \subseteq S \times S$ are **transitions** such that:
 - $(\ell, \nu) \rightarrow (\ell', \nu')$ if $\ell \xrightarrow{g,r} \ell'$, $\nu \models g$, $\nu' = \nu[r]$,
 - $(\ell, \nu) \rightarrow (\ell, \nu + \delta)$ for all $\delta \in \mathbb{R}_{\geq 0}$ such that $\nu + t \models I(\ell)$ for any $0 \leq t \leq \delta$.

Semantics in terms of transition system (S, s_0, \rightarrow) with:

- $S = \{(\ell, \nu) \mid \ell \in L, \nu : X \rightarrow \mathbb{R}_{\geq 0}, \nu \models I(\ell)\}$ are the **states**,
- $s_0 = (\ell_0, \nu_0)$, where $\nu_0(x) = 0$ for any x (**initial state**),
- $\rightarrow \subseteq S \times S$ are **transitions** such that:
 - $(\ell, \nu) \rightarrow (\ell', \nu')$ if $\ell \xrightarrow{g,r} \ell'$, $\nu \models g$, $\nu' = \nu[r]$,
 - $(\ell, \nu) \rightarrow (\ell, \nu + \delta)$ for all $\delta \in \mathbb{R}_{\geq 0}$ such that $\nu + t \models I(\ell)$ for any $0 \leq t \leq \delta$.

Infinite!

Zones are sets of clock valuations that satisfy a clock constraint –
 $Z = \{v \in \mathbb{R}_{\geq 0}^X \mid v \models g\}$.

Operations on zones:

- $Z \wedge g = \{v \in Z \mid v \models g\}$,
- $Z^\uparrow = \{v + \delta \mid v \in Z, \delta \in \mathbb{R}_{\geq 0}\}$,
- $Z[r] = \{v[r] \mid v \in Z\}$.

Successor relation on **zone graph**:

- $(\ell, Z) \rightsquigarrow (\ell, Z^\uparrow \wedge I(\ell))$ – delay successor,
- $(\ell, Z) \rightsquigarrow (\ell', (Z \wedge g)[r] \wedge I(\ell'))$ if $\ell \xrightarrow{g,r} \ell'$ – discrete successor.

The part reachable from (ℓ_0, Z_0) can be made **finite**.

Zones are usually implemented as **Difference Bound Matrices**.

Reachability in zone graph

```
1: Waiting :=  $\{(\ell_0, Z_0)\}$ 
2: Passed :=  $\emptyset$ 
3: while Waiting  $\neq \emptyset$  do
4:   Choose and remove  $(\ell, Z)$  from Waiting
5:   if  $\ell = s$  and  $Z \cap \varphi \neq \emptyset$  then
6:     return TRUE
7:   end if
8:   if  $Z \not\subseteq Z'$  for all  $(\ell, Z') \in \textit{Passed}$  then
9:     Passed := Passed  $\cup \{(\ell, Z)\}$ 
10:    Waiting := Waiting  $\cup \{(\ell', Z') \mid (\ell, Z) \rightsquigarrow (\ell', Z') \wedge Z' \neq \emptyset\}$ 
11:  end if
12: end while
13: return FALSE
```

Reachability in zone graph

```
1: Waiting :=  $\{(\ell_0, Z_0)\}$ 
2: Passed :=  $\emptyset$ 
3: while Waiting  $\neq \emptyset$  do
4:   Choose and remove  $(\ell, Z)$  from Waiting
5:   if  $\ell = s$  and  $Z \cap \varphi \neq \emptyset$  then
6:     return TRUE
7:   end if
8:   if  $Z \not\subseteq Z'$  for all  $(\ell, Z') \in \textit{Passed}$  then
9:     Passed := Passed  $\cup \{(\ell, Z)\}$ 
10:    Waiting := Waiting  $\cup \{(\ell', Z') \mid (\ell, Z) \rightsquigarrow (\ell', Z') \wedge Z' \neq \emptyset\}$ 
11:  end if
12: end while
13: return FALSE
```

We need to do following things with a zone:

- Decide if it satisfies φ .
- Decide whether it is subset of another one.
- Compute successors $(Z \wedge g, Z^\uparrow, Z[r])$.

Various real-time model-checking tools, e.g. UPPAAL.

Used for verification of

- Car/airplane controllers
- Protocols – audio/video transfer
- Other, where real time matters

$$(\mathbb{R}_{\max}, \oplus, \otimes) = (\mathbb{R} \cup \{-\infty\}, \max, +)$$

- associative, commutative, distributive
- \oplus is idempotent as $a \oplus a = a$
- \oplus does not have inverse elements s.t. $a \oplus b = -\infty$
- **max-plus semiring**

We can extend \oplus and \otimes to vectors and matrices in the usual way.

Analogues to **convex polyhedra** in ordinary euclidean space.

Definition

A convex max-plus polyhedron is a subset of \mathbb{R}_{max}^n that satisfies a finite set of max-plus linear inequalities.

Analogues to **convex polyhedra** in ordinary euclidean space.

Definition

A convex max-plus polyhedron is a subset of \mathbb{R}_{max}^n that satisfies a finite set of max-plus linear inequalities.

(demonstration)

Analogues to **convex polyhedra** in ordinary euclidean space.

Definition

A convex max-plus polyhedron is a subset of \mathbb{R}_{max}^n that satisfies a finite set of max-plus linear inequalities.

(demonstration)

How can we finitely represent them?

Representing max-plus polyhedron by finite **set of linear inequalities** of the form $\mathbf{ax} \oplus \mathbf{b} \geq \mathbf{cx} \oplus \mathbf{d}$. A max-plus polyhedron can be described by two matrices A, C and two vectors \mathbf{b}, \mathbf{d} :

$$P = \{\mathbf{x} \mid \mathbf{Ax} \oplus \mathbf{b} \geq \mathbf{Cx} \oplus \mathbf{d}\}$$

Unlike with ordinary polyhedra, we can use equalities as well as inequalities.

Max-plus polyhedron can be also represented in terms of its **extreme points** (or **generators**). Let $\text{co}(V)$ be the set of all **convex combinations** of points in V and $\text{cone}(W)$ the set of all **linear combinations** of points in W .

Every max-plus polyhedron can be described as

$$P = \text{co}(V) \oplus \text{cone}(W)$$

i.e. every point can be written as

$$\alpha_1 \mathbf{v}_1 \oplus \cdots \oplus \alpha_p \mathbf{v}_p \oplus \beta_1 \mathbf{w}_1 \oplus \cdots \oplus \beta_q \mathbf{w}_q$$

for $\alpha_1 \oplus \cdots \oplus \alpha_p = \tilde{1}$ and some $\beta_1 \dots \beta_q$.

We can also represent n -dimensional max-plus polyhedron as a $(n + 1)$ -dimensional max-plus cone using homogeneous coordinates. If $P = \text{co}(V) \oplus \text{cone}(W)$ is max-plus polyhedron, we can define

$$Z = \{(\mathbf{v}, \tilde{\mathbf{1}}) \mid \mathbf{v} \in V\} \cup \{(\mathbf{w}, \tilde{\mathbf{0}}) \mid \mathbf{w} \in W\}.$$

It can be seen that

$$P = \{\mathbf{x} \mid (\mathbf{x}, \tilde{\mathbf{1}}) \in \text{cone}(Z)\}.$$

This representation allows us to use much simpler algorithms as we don't have to distinguish two kinds of extreme points.

We want to use max-plus polyhedra as a data structure for manipulating zones.

- Every zone is also a max-plus polyhedron.
- Can operations on MPPs be faster than on DBMs?
- **Convex union** overapproximation is more precise for MPPs than for DBMs.

Does the equation $Gy = x$ have a solution? We cannot solve it directly as we don't have the inverse elements.

However, we know how to find the maximal solution to $G\hat{y} \leq x$ (always exists). We do this and then check whether $G\hat{y} = x$.

Using this algorithm, we can check whether one MPP is **subset** of another. We also use it for **removal of redundant generators**.

Delay – we just add all of the convex generators to the cone generators:

$$W := W \cup V$$

Reset:

```
for all  $v \in V$  do  
   $v_j := 0$   
end for  
for all  $w \in W$  do  
   $w_j := -\infty$   
end for
```

Convex union of two polyhedra can be obtained by taking an union of their extreme points.

| | Max-plus polyhedra | DBM |
|-------------------------------|--------------------|----------|
| Emptiness test | $O(1)$ | $O(1)$ |
| Inclusion test | $O(p^2 n)$ | $O(n^2)$ |
| Constraint satisfaction | $O(p^2 n)$ | $O(n^2)$ |
| Constraint intersection | $O(p^2 n)$ | $O(n^2)$ |
| Delay | $O(pn)$ | $O(n)$ |
| Backward delay | $O(p^2 n^2)$ | $O(n^2)$ |
| Resetting clocks | $O(p)$ | $O(n)$ |
| Removing constraints | $O(p + n)$ | $O(n)$ |
| Union overapproximation | $O(pn)$ | $O(n^2)$ |
| Removing redundant generators | $O(p^2 n)$ | $O(n^3)$ |

Table: Complexity of algorithms for max-plus polyhedra and DBMs. Here, n denotes the number of clocks and p the number of generators of the polyhedron.

Future work:

- Strict constraints
- Performance
- Federation data structure

Summary:

- Timed automata as a formalism for real-time system
- Max-plus polyhedra can be used to implement symbolic states for TA
- MPPs can express some non-convex properties without disjunction
- Not (yet) suited to replace DBMs