Liveness in L/U-Parametric Timed Automata

Étienne André and Didier Lime

Université Paris 13, LIPN and École Centrale de Nantes, LS2N

ACSD, 28th of June 2017, Zaragoza, Spain
Parametric timed automata (PTA) allow for flexible, abstract, and robust modelling;

The answer to parametric model-checking is appealing;

Many undecidability results exist for safety / reachability properties;

And a few decidable subclasses:

- L/U PTA [HRSV02];
- IP-PTA [ALR16];
- bounded integer PTA [JLR15].
Parametric timed automata (PTA) allow for flexible, abstract, and robust modelling;

The answer to **parametric** model-checking is appealing;

Many undecidability results exist for safety / reachability properties;

And a few decidable subclasses:

- L/U PTA [HRSV02];
- IP-PTA [ALR16];
- bounded integer PTA [JLR15].

What about **liveness**?
Parametric Timed Automata [AHV93]

\[ x = p_1 \]
\[ a \]
\[ x := 0 \]

\[ x = 0 \land y \leq p_2, \ b \]

\[ y \leq p_2 \]
Parametric Timed Automata [AHV93]

$x = p_1$

$a$

$x := 0$

For $p_1 = 1.2$ and $p_2 = 4$:

$\ell_0$

$x = 0 \xrightarrow{1.2} x = 1.2$

$\ell_0$

$y = 0 \xrightarrow{a} y = 1.2$

$\ell_0$

$y = 0 \xrightarrow{b} x = 0$

$\ell_1$

$\ell_1$

$x = 0 \xrightarrow{2.4} x = 2.4$

$y = 1.2 \xrightarrow{b} y = 1.2$

$y = 1.2 \xrightarrow{2.4} y = 3.6$
Parameters are used either as lower bounds or as upper bounds, never both.

- **Monotonicity**: increasing upper bounds or decreasing lower bounds gives more behaviours.
Liveness in (Parametric) Timed Automata

- Our liveness properties concern **maximal** paths:
  - Existence of an **infinite** maximal path (discrete **cycle**, denoted EC);
  - Existence of a **finite** maximal path (deadlock, denoted ED);
  - Existence of a maximal path preserving some property (CTL **EG** property).
Liveness in (Parametric) Timed Automata

- **Our liveness properties concern maximal paths:**
  - Existence of an infinite maximal path (discrete cycle, denoted EC);
  - Existence of a finite maximal path (deadlock, denoted ED);
  - Existence of a maximal path preserving some property (CTL $\text{EG}$ property).

- **Parametric properties:**
  - $\phi$-emptiness: is the set of parameter valuations s.t. $\phi$ holds empty?
  - $\phi$-universality: is the set of parameter valuations s.t. $\phi$ holds universal?
Liveness in (Parametric) Timed Automata

- Our **liveness** properties concern **maximal** paths:
  - Existence of an **infinite** maximal path (discrete **cycle**, denoted EC);
  - Existence of a **finite** maximal path (deadlock, denoted ED);
  - Existence of a maximal path preserving some property (CTL **EG** property).

- **Parametric** properties:
  - $\phi$-emptiness: is the set of parameter valuations s.t. $\phi$ holds empty?
  - $\phi$-universality: is the set of parameter valuations s.t. $\phi$ holds universal?
### Results from the Literature

<table>
<thead>
<tr>
<th>Class</th>
<th>PTA</th>
<th>L/U PTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>EC-emptiness</td>
<td>open</td>
<td>PSPACE-c.(^1)</td>
</tr>
<tr>
<td>ED-emptiness</td>
<td>open</td>
<td>open</td>
</tr>
<tr>
<td>EG-emptiness</td>
<td>open</td>
<td>open</td>
</tr>
</tbody>
</table>

\(^1\)Integer parameters [BL09].
EC-emptiness is PSPACE-c for L/U PTAs

- There exists a **rational** parameter valuation s.t. there is a cycle iff there exists an **integer** valuation.
- Use the **monotonicity** property of L/U PTAs: **round** up for upper bounds, down for lower bounds to get a good **integer** valuation.
EC-emptiness is undecidable for PTAs

- Reduce from the counter boundedness problem of 2-counter machines
  - Finite-state machine + 2 non-negative integer counters;
  - increment some counter and go to some state;
  - if some counter is zero then decrement it and go to some state; otherwise go to some other state;
EC-emptiness is undecidable for PTAs

- Reduce from the counter boundedness problem of 2-counter machines
  - Finite-state machine + 2 non-negative integer counters;
  - increment some counter and go to some state;
  - if some counter is zero then decrement it and go to some state; otherwise go to some other state;
- States of the machines are encoded by locations $q_i$;
- Counters are encoded by clocks $y, z$ and one parameter $p$: when clock $x$ is null,
  \[
  y = 1 - c_1 p \\
  z = 1 - c_2 p
  \]
EC-emptiness is undecidable for PTAs

- Reduce from the counter boundedness problem of 2-counter machines
  - Finite-state machine + 2 non-negative integer counters;
  - increment some counter and go to some state;
  - if some counter is zero then decrement it and go to some state; otherwise go to some other state;
- States of the machines are encoded by locations \( q_i \);
- Counters are encoded by clocks \( y, z \) and one parameter \( p \): when clock \( x \) is null,
  \[
  y = 1 - c_1 p \\
  z = 1 - c_2 p
  \]
- Initialisation:
  \[
  x = p \land x > 0 \\
  x = 1 \\
  x ::= 0
  \]
EC-emptiness is undecidable for PTAs

- Increment:

\[
\begin{align*}
z &= 1 \\
z &:= 0 \\
x &= 0 \\
q_i &\rightarrow l_{i1} \\
y &= p + 1 \\
y &:= 0 \\
l_{i1} &\rightarrow l_{i2} \\
x &= 1 \\
x &:= 0 \\
l_{i2} &\rightarrow l_{i3} \\
y &= p + 1 \\
y &:= 0 \\
l_{i3} &\rightarrow l'_{i2} \\
z &= 1 \\
z &:= 0 \\
l'_{i2} &\rightarrow q_j \\
y &= 1 \\
y &:= 0 \\
q_j &\rightarrow l'_{i2} \\
z &= 1 \\
z &:= 0 \\
\end{align*}
\]
EC-emptiness is undecidable for PTAs

- Increment:

```
EC-emptiness is undecidable for PTAs

Increment:

- \(q_i\) to \(l_{i1}\):
  - \(x = 0\)
  - \(y = 1 - c_1 p\)
  - \(z = 1 - c_2 p\)

- \(l_{i1}\) to \(l_{i2}\):
  - \(x = 0\)
  - \(y = 1 - c_1 p\)
  - \(z = 1 - c_2 p\)

- \(l_{i2}\) to \(l_{i3}\):
  - \(x = c_2 p\)
  - \(y = 1 - (c_1 - c_2) p\)
  - \(z = 0\)

- \(l_{i3}\) to \(q_j\):
  - \(x = (c_1 + 1) p\)
  - \(y = 0\)
  - \(z = (c_1 - c_2 + 1) p\)

- \(q_j\) to \(l_{i2}\):
  - \(x = 0\)
  - \(y = 1 - (c_1 + 1) p\)
  - \(z = 1 - c_2 p\)
```
EC-emptiness is undecidable for PTAs

- **Increment:**

  \[
  \begin{align*}
  q_i & \xrightarrow{0} l_{i1} & x = 0 & \quad x = 0 & \quad x = 0 & \quad x = 0 \\
  & & y = 1 - c_1 p & \quad y = 1 - c_1 p & \quad y = 1 - (c_1 - c_2) p & \quad y = 1 - (c_1 + 1) p \\
  & & z = 1 - c_2 p & \quad z = 1 - c_2 p & \quad z = 0 & \quad z = 1 - c_2 p \\
  \end{align*}
  \]

  \[
  \begin{align*}
  l_{i1} & \xrightarrow{c_2 p} l_{i2} & y = p + 1 & \quad y = p + 1 & \quad y = p + 1 \\
  & & y = 0 & \quad y = 0 & \quad y = 0 \\
  & & z = 1 & \quad z = 1 & \quad z = 1 \\
  \end{align*}
  \]

  \[
  \begin{align*}
  l_{i2} & \xrightarrow{(c_1 - c_2 + 1)p} l_{i3} & x = 0 & \quad x = 0 & \quad x = 0 \\
  & & y = 1 - (c_1 - c_2) p & \quad y = 1 - (c_1 + 1) p & \quad y = 1 - (c_1 + 1) p \\
  & & z = 0 & \quad z = 1 - c_2 p & \quad z = 1 - c_2 p \\
  \end{align*}
  \]

  \[
  \begin{align*}
  l_{i3} & \xrightarrow{(c_1 + 1)p} q_j & x = 0 & \quad x = 0 & \quad x = 0 \\
  & & y = 0 & \quad y = 1 - (c_1 + 1) p & \quad y = 1 - (c_1 + 1) p \\
  & & z = (c_1 - c_2 + 1)p & \quad z = 1 - c_2 p & \quad z = 1 - c_2 p \\
  \end{align*}
  \]

- **implies** \( p \leq \frac{1}{c_1+1} \) otherwise it **blocks** at \( l_{i3} \).
EC-emptiness is undecidable for PTAs

- Zero-test and decrement:

  - $x = 0$
  - $y < 1$
  - $y = 1$
  - $x = p + 1$
  - $z = p + 1$

- $c_1 = 0$ iff $y = 1$. 
- Decrement is similar to increment.
EC-emptiness is undecidable for PTAs

- Halting:

  There is a (discrete) cycle in the PTA iff the counter are bounded:
  - if the machine halts, $q_{halt}$ is reachable $\rightarrow$ cycle;
  - if the machine does not halt but the counters are bounded, there is a parameter valuation small enough to have a cycle among the instruction widgets;
  - if the counters are unbounded, for any valuation, the PTA will eventually block in the increment widget.
EC-emptiness is undecidable for PTAs

- **Halting:**

  - There is a (discrete) cycle in the PTA iff the counter are bounded:
    - if the machine halts, \( q_{\text{halt}} \) is reachable \( \rightarrow \) cycle;
    - if the machine does not halt but the counters are bounded, there is a parameter valuation **small enough** to have a cycle among the instruction widgets;
    - if the counters are unbounded, for any valuation, the PTA will eventually **block** in the increment widget.
ED-emptiness is undecidable for L/U PTAs

- Reduce from the **halting** problem of 2-counter machines;
ED-emptiness is undecidable for L/U PTAs

- Reduce from the **halting** problem of 2-counter machines;
- Change previous construction to “split” parameters and get an L/U PTA:

\[
\begin{align*}
q_i &\xrightarrow{x=0} l_{i1} & l_{i2} &\xrightarrow{p^- + 1 \leq y \leq p^+ + 1} q_j \\
q_i &\xrightarrow{p^- + 1 \leq y \leq p^+ + 1} l_{i2} & l_{i3} &\xrightarrow{z = 1} q_j \\
\end{align*}
\]

- We use the deadlock property to enforce \( p^- = p^+ \).
ED-emptiness is undecidable for L/U PTAs

- Initialisation, enforce $p^- \leq p^+$:
  
  $$
  \begin{align*}
  p^- \leq x \leq p^+ \\
  x, y, z := 0 \\
  x := 0
  \end{align*}
  $$

- Halting, there is a **deadlock** in $q_{\text{halt}}$ iff $p^+ \leq p^-$ (and $p^- > 0$):
  
  $$
  p^- \leq x < p^+ \\
  p^- \leq x \land x = 0
  $$

- Add a transition with guard true from all locations but $q_{\text{halt}}$;
- the machine **halts** iff there exists a valuation such that $p^- = p^+$ and there is a **deadlock** in the PTA.
EG-emptiness is undecidable for L/U PTAs

- by reduction from the **halting** problem of 2-counter machines;
- similar to the ED-construction with a different encoding adapted from [BBLS15];
- the main idea is to eliminate cycles by:
  - making sure all widgets execute in 1 t.u.;
  - add a global invariant limiting the **total execution time** so that it does not exceed some parameter \( p_2 \);
  - then the PTA can only execute **at most** \( p_2 \) instructions and \( p_2 \) has to be **big enough** for executing a halting sequence.
### Results up to now

<table>
<thead>
<tr>
<th>Class</th>
<th>PTA</th>
<th>L/U PTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>EC-emptiness</td>
<td>Undec.</td>
<td>PSPACE-c.</td>
</tr>
<tr>
<td>ED-emptiness</td>
<td>Undec.</td>
<td>Undec.</td>
</tr>
<tr>
<td>EG-emptiness</td>
<td>Undec.</td>
<td>Undec.</td>
</tr>
</tbody>
</table>

We can find some decidability by considering parameters are bounded (each takes its values in some bounded interval); changes nothing for PTAs; we consider both (topologically) closed and open parameter domains.
Bounded parameters

Results up to now

<table>
<thead>
<tr>
<th>Class</th>
<th>PTA</th>
<th>L/U PTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>EC-emptiness</td>
<td>Undec.</td>
<td>PSPACE-c.</td>
</tr>
<tr>
<td>ED-emptiness</td>
<td>Undec.</td>
<td>Undec.</td>
</tr>
<tr>
<td>EG-emptiness</td>
<td>Undec.</td>
<td>Undec.</td>
</tr>
</tbody>
</table>

- We can find some decidability by considering parameters are **bounded** (each takes its values in some bounded interval);
- Changes nothing for **PTAs**;
- We consider both (topologically) **closed** and **open** parameter domains.
EG-emptiness is decidable for closed bounded L/U PTA

1. Test if there is an infinite path preserving $\phi$ in the TA obtained by setting:
   - lower bounds to their minimum value,
   - and upper bounds to their maximal values.
   i.e. verify CTL property “EG ($\phi \land \text{EX true}$)” on the region graph of the TA.
2. if yes we are done
EG-emptiness is decidable for closed bounded L/U PTA

1. Test if there is an infinite path preserving $\phi$ in the TA obtained by setting:
   - lower bounds to their minimum value,
   - and upper bounds to their maximal values.
   i.e. verify CTL property “$\text{EG} (\phi \land \text{EX} \text{ true})$” on the region graph of the TA.

2. if yes we are done

3. otherwise all paths preserving $\phi$ are finite: explore them symbolically, using the symbolic polyhedral abstraction of linear hybrid automata;

4. test all symbolic states on those paths for deadlocks:
   - consider all states that can reach some guard (classic past operator)
   - check if those states cover the whole symbolic state (polyhedral union and inclusion).
EG-emptiness is undecidable for open bounded L/U PTA

- Reduce from the halting problem of 2-counter machines
- Make sure all widgets execute in \([p_2^-, p_2^+]\) t.u. (instead of 1);

\[
\begin{align*}
p_2^- & \leq z \leq p_2^+ \\
z & := 0
\end{align*}
\]

\[
\begin{align*}
p_1^- + p_2^- & \leq y \\
y & := 0
\end{align*}
\]

- use the open parameter domain to enforce \(p_2^- > 0\);
- add a global invariant so that the whole PTA can only execute for 1 t.u. to eliminate cycles;
- the machine halts iff there exists a parameter valuation s.t. \(p_1^- = p_1^+\) and \(p_2^- = p_2^+\) and there is a deadlock in the PTA.
The other results follow directly from the previous constructions;

We conjecture that EC-emptiness for open bounded L/U PTAs is **decidable** with techniques similar to [San11].
Conclusion and Perspectives

Summary:
- We have exhibited a very thin border of decidability for liveness properties;
- It depends on the boundedness of the parameters and the topological closure of their initial domain.

Future work:
- Prove that EC-emptiness for open bounded LU PTAs is decidable;
- Complete the results for the universality problems;
- Find the complexity of EG-emptiness for closed bounded L/U PTA.
References

- **Rajeev Alur, Thomas A. Henzinger, and Moshe Y. Vardi.**
  Parametric real-time reasoning.

- **Étienne André, Didier Lime, and Olivier H. Roux.**
  Decision problems for parametric timed automata.

- **Nikola Beneš, Peter Bezděk, Kim G. Larsen, and Jiří Srba.**
  Language emptiness of continuous-time parametric timed automata.

- **Laura Bozzelli and Salvatore La Torre.**
  Decision problems for lower/upper bound parametric timed automata.

- **Thomas Hune, Judi Romijn, Mariëlle Stoelinga, and Frits W. Vaandrager.**
  Linear parametric model checking of timed automata.

- **Aleksandra Jovanović, Didier Lime, and Olivier H. Roux.**
  Integer parameter synthesis for timed automata.