Weighted Timed Games: Positive Results with Negative Costs

Benjamin Monmege
Université Libre de Bruxelles, Belgium

Thomas Brihaye (UMons)
Gilles Geeraerts (ULB)
Shankara Krishna (IITB)
Lakshmi Manasa (IITB)
Ashutosh Trivedi (IITB)

1st Cassting Workshop ETAPS 2014

April 12, 2014
Smart Houses on a Grid (Jadevej Case)

Eight houses  
Electric local grid  
Each house:
  ▶ Solar panels  
  ▶ Electric heating  
  ▶ Storage of energy
Smart Houses on a Grid (Jadevej Case)

Eight houses
Electric local grid

Each house:
- Solar panels
- Electric heating
- Storage of energy

**Goal**: for each house, optimize its behavior to reduce its energy bill

How to compute the expenses of a house?
Smart Houses on a Grid (Jadevej Case)

Eight houses
Electric local grid

Each house:
- Solar panels
- Electric heating
- Storage of energy

**Goal**: for each house, optimize its behavior to reduce its energy bill

How to compute the expenses of a house?

<table>
<thead>
<tr>
<th>Solar panel ON</th>
<th>Solar panel OFF</th>
<th>Solar panel OFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selling energy: $+2\text{€/t.u.}$</td>
<td>Selling energy: $+2\text{€/t.u.}$</td>
<td>Selling energy: $+1\text{€/t.u.}$</td>
</tr>
<tr>
<td>Consumption: $0\text{€/t.u.}$</td>
<td>Consumption: $-2\text{€/t.u.}$</td>
<td>Consumption: $-1\text{€/t.u.}$</td>
</tr>
<tr>
<td>Storing energy: $0\text{€/t.u.}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

+ fixed cost to start selling or buying energy
Smart Houses on a Grid (Jadevej Case)

Eight houses
Electric local grid

Each house:
- Solar panels
- Electric heating
- Storage of energy

**Goal**: for each house, optimize its behavior to reduce its energy bill

How to compute the expenses of a house?

- **Solar panel ON**
  - Selling energy: $+2\text{€/t.u.}$
  - Consumption: $0\text{€/t.u.}$
  - Storing energy: $0\text{€/t.u.}$

- **Solar panel OFF**
  - Selling energy: $+2\text{€/t.u.}$
  - Consumption: $+2\text{€/t.u.}$
  - Storing energy: $0\text{€/t.u.}$

+ fixed cost to start selling or buying energy

Our contribution: Synthesize optimal behaviors in each phase by solving weighted timed timed games with a limited number of distinct rates
Weighted Timed Games

Timed Automaton with partition of states between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of infinite game with weights

\[
\begin{align*}
\ell_1 & : x \leq 1, x := 0, 0 \\
\ell_2 & : x > 0 \quad \text{[}x \leq 2\text{]} \\
\ell_3 & : x \leq 2, 0 \\
\ell_4 & : x \leq 1, 1 \\
\ell_5 & : x \geq 1, x := 0, 0 \\
\ell_6 & : x \geq 1, 2
\end{align*}
\]

Weight of a play:
\[
\{+\infty \text{ if not reached, total payoff otherwise}\}
\]
Weighted Timed Games

Timed Automaton
with partition of states
between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of
infinite game with weights

Weight of a play:
{+∞ if not reached
 otherwise total payoff

(ℓ₁, 0)
Weighted Timed Games

Timed Automaton with partition of states between 2 players
+ reachability objective
+ rates in locations
+ costs over transitions

Semantics in terms of infinite game with weights

\[(\ell_1, 0) \xrightarrow{0.4} (\ell_4, 0.4)\]
Weighted Timed Games

Timed Automaton with partition of states between 2 players + reachability objective + rates in locations + costs over transitions

Semantics in terms of infinite game with weights

\[(\ell_1, 0) \xrightarrow{0.4, \searrow} (\ell_4, 0.4) \xrightarrow{0.6, \rightarrow} (\ell_5, 0)\]
Weighted Timed Games

Timed Automaton with partition of states between 2 players + reachability objective + rates in locations + costs over transitions

Semantics in terms of infinite game with weights

\[(\ell_1, 0) \xrightarrow{0.4} (\ell_4, 0.4) \xrightarrow{0.6} (\ell_5, 0) \xleftarrow{1.5} (\ell_4, 0) \xrightarrow{1.1} (\ell_5, 0) \xrightarrow{2, \cdot} (\checkmark, 2)\]
Weighted Timed Games

Timed Automaton with partition of states between 2 players + reachability objective + rates in locations + costs over transitions

Semantics in terms of infinite game with weights

\[
\begin{align*}
(l_1, 0) & \xrightarrow{0.4} (l_4, 0.4) & & \xrightarrow{0.6} (l_5, 0) & & \xleftarrow{1.5} (l_4, 0) & & \xrightarrow{1.1} (l_5, 0) & & \xrightarrow{2} (\checkmark, 2) & \approx 3.8
\end{align*}
\]
Weighted Timed Games

Timed Automaton with partition of states between 2 players + reachability objective + rates in locations + costs over transitions

Semantics in terms of infinite game with weights

$\begin{align*}
(x > 0, x := 0, 0) & \xrightarrow{\ell_2} (x \leq 2, x := 0, 0) \\
(x \leq 1, x := 0, 0) & \xrightarrow{\ell_1} (x \leq 1, x := 0, 0) \\
(x \leq 1, x := 0, 0) & \xrightarrow{\ell_3} (x \leq 2, x := 0, 0) \\
(x \leq 2, x := 0, 0) & \xrightarrow{\ell_4} (x \leq 2, x := 0, 0) \\
(x \leq 2, x := 0, 0) & \xrightarrow{\ell_5} (x \leq 2, x := 0, 0) \\
(x \leq 2, x := 0, 0) & \xrightarrow{\ell_6} (x \leq 2, x := 0, 0) \\
(x \leq 2, x := 0, 0) & \xrightarrow{\ell_6} (x \leq 2, x := 0, 0)
\end{align*}$

\[ (\ell_1, 0) \xrightarrow{0.4} (\ell_4, 0.4) \xrightarrow{0.6} (\ell_5, 0) \xrightarrow{1.5} (\ell_4, 0) \xrightarrow{1.1} (\ell_5, 0) \xrightarrow{2} (\checkmark, 2) \]
\[ 0.4 + 1 - 3 \times 0.6 + 1.5 - 3 \times 1.1 + 2 \times 2 + 2 = 3.8 \]

\[ (\ell_1, 0) \xrightarrow{0.2} (\ell_2, 0) \xrightarrow{0.9} (\ell_3, 0.9) \xrightarrow{0.2} (\ell_3, 0) \xrightarrow{0.9} (\ell_3, 0) \]
\[ 0.2 + 0.9 - 0.2 - 0.9 = +\infty (\checkmark \text{ not reached}) \]
Weighted Timed Games

Timed Automaton with partition of states between 2 players + reachability objective + rates in locations + costs over transitions

Semantics in terms of infinite game with weights

Weight of a play: \[
\begin{cases}
+\infty & \text{if } \checkmark \text{ not reached} \\
total \text{ payoff} & \text{otherwise}
\end{cases}
\]
Strategy for each player: mapping of finite runs to a delay and an action
Strategies and objectives

Strategy for each player: mapping of finite runs to a delay and an action

Goal of player ◯: reach ✓ and minimize the cost
Goal of player □: avoid ✓ or, if not possible, maximize the cost
Strategies and objectives

Strategy for each player: mapping of finite runs to a delay and an action

Goal of player $\bigcirc$: reach $\checkmark$ and minimize the cost
Goal of player $\square$: avoid $\checkmark$ or, if not possible, maximize the cost

Main object of interest:
\[
\overline{\text{Val}}(\ell, v) = \inf_{\sigma_\bigcirc \in \text{Strat}_\bigcirc} \sup_{\sigma_\square \in \text{Strat}_\square} \text{Wt}(\text{Play}((\ell, v), \sigma_\bigcirc, \sigma_\square)) \in \mathbb{R} \cup \{-\infty, +\infty\}
\]

What player $\bigcirc$ can guarantee as a payoff? and design good strategies
State of the art

Decision problem: does there exist a strategy for player \( \bigcirc \) ensuring a weight not greater than a given constant?

▶ One-player case (Weighted timed automata): optimal reachability problem is PSPACE-complete
▶ Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
▶ and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdzinski, 2013, Haase, Ouaknine, and Worrell, 2012]
▶ Weighted timed (two-player) games are undecidable [Bouyer, Brihaye, and Markey, 2006a], even with only non-negative weights and 3 clocks

This talk: One-clock weighted timed games with negative weights
State of the art

Decision problem: does there exist a strategy for player $\bigcirc$ ensuring a weight not greater than a given constant?

- One-player case (Weighted timed automata): optimal reachability problem is PSPACE-complete
  - Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
  - and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdziński, 2013, Haase, Ouaknine, and Worrell, 2012]
State of the art

Decision problem: does there exist a strategy for player $\bigcirc$ ensuring a weight not greater than a given constant?

- One-player case (**Weighted timed automata**): optimal reachability problem is **PSPACE-complete**
  - Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
  - and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdziński, 2013, Haase, Ouaknine, and Worrell, 2012]

- Weighted timed (two-player) games are **undecidable** [Bouyer, Brihaye, and Markey, 2006a], even with only non-negative weights and 3 clocks
State of the art

Decision problem: does there exist a strategy for player $\bigcirc$ ensuring a weight not greater than a given constant?

- **One-player case** (*Weighted timed automata*): optimal reachability problem is **PSPACE-complete**
  - Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
  - and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdziński, 2013, Haase, Ouaknine, and Worrell, 2012]

- **Weighted timed (two-player) games** are **undecidable** [Bouyer, Brihaye, and Markey, 2006a], even with only non-negative weights and 3 clocks

State of the art

Decision problem: does there exist a strategy for player □ ensuring a weight not greater than a given constant?

- One-player case (Weighted timed automata): optimal reachability problem is PSPACE-complete
  - Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
  - and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdziński, 2013, Haase, Ouaknine, and Worrell, 2012]

- Weighted timed (two-player) games are undecidable [Bouyer, Brihaye, and Markey, 2006a], even with only non-negative weights and 3 clocks


State of the art

Decision problem: does there exist a strategy for player $\Diamond$ ensuring a weight not greater than a given constant?

- One-player case (Weighted timed automata): optimal reachability problem is PSPACE-complete
  - Algorithm based on region abstraction [Bouyer, Brihaye, Bruyère, and Raskin, 2007];
  - and hardness shown for timed automata with at least 2 clocks [Fearnley and Jurdziński, 2013, Haase, Ouaknine, and Worrell, 2012]
- Weighted timed (two-player) games are undecidable [Bouyer, Brihaye, and Markey, 2006a], even with only non-negative weights and 3 clocks

This talk: One-clock weighted timed games with negative weights
Why things are complex with negative weights? (even in weighted untimed finite games)

- Value $-\infty$: detection is as hard as mean-payoff. No hope for complexity better than \( \text{NP} \cap \text{co-NP} \), or pseudo-polynomial

- Memory complexity
  - Player \( \bigcirc \) needs memory

- Player \( \square \) needs infinite memory in weighted timed games
One-clock Binary Weighted Timed Games (1BWTG)

Assumption: rates of locations \(\{p^−, p^+\}\) included in \(\{0, +d, −d\}\) \((d \in \mathbb{N})\) (no assumption on weights of transitions)

\[
\begin{align*}
&x < 1, x := 0, 0 \\
x > 0, &\ l_2 \rightarrow 1 \quad \text{[}x \leq 2\text{]} \\
x := 0, 0, &\ l_3 \rightarrow −1 \quad \text{[}x \leq 2\text{]} \quad x > 1, 1 \\
x \geq 1, &\ l_4 \rightarrow −1 \quad \text{[}x \leq 2\text{]} \\
x := 0, 0, &\ l_5 \rightarrow 1 \quad \text{[}x \geq 1\text{]} \quad x \geq 1, 2 \\
x \geq 1, &\ l_6 \rightarrow √ \\
x \leq 1, 1, &\ l_1 \rightarrow 1 \\
x := 0, 0, &\ l_2 \rightarrow 1 \\
x := 0, 0, &\ l_3 \rightarrow −1 \\
x := 0, 0, &\ l_5 \rightarrow 1
\end{align*}
\]

Results

Theorem:
- Computation of the value \( \overline{\text{Val}}(\ell, v) \) of states of a 1BWTG in pseudo-polynomial time
- Synthesis of \( \varepsilon \)-optimal strategies for player \( \bigcirc \) in pseudo-polynomial time

Theorem: Non-negative case

In case of a 1BWTG with only non-negative weights, all complexities drop down to polynomial.
First idea: symetrize the point of view

Value for player $\bigcirc$: $\overline{\text{Val}}(\ell, v) = \inf_{\sigma_\bigcirc \in \text{Strat}_\bigcirc} \sup_{\sigma_\square \in \text{Strat}_\square} Wt(\text{Play}(\ell, v, \sigma_\bigcirc, \sigma_\square))$

Value for player $\square$: $\text{Val}(\ell, v) = \sup_{\sigma_\square \in \text{Strat}_\square} \inf_{\sigma_\bigcirc \in \text{Strat}_\bigcirc} Wt(\text{Play}(\ell, v, \sigma_\bigcirc, \sigma_\square))$

How to compare them? $\text{Val}(\ell, v) \leq \overline{\text{Val}}(\ell, v)$

Theorem: (continued)

▶ $1BWTGs$ are determined: $\text{Val}(\ell, v) = \overline{\text{Val}}(\ell, v)$

▶ Synthesis of $\varepsilon$-optimal strategies for player $\square$ in pseudo-polynomial time (and polynomial in case of non-negative weights)
First idea: symetrize the point of view

Value for player $\bigcirc$: $\overline{\text{Val}}(\ell, v) = \inf_{\sigma_\bigcirc \in \text{Strat}_\bigcirc} \sup_{\sigma_\square \in \text{Strat}_\square} \text{Wt}(\text{Play}(\ell, v, \sigma_\bigcirc, \sigma_\square))$

Value for player $\square$: $\underline{\text{Val}}(\ell, v) = \sup_{\sigma_\square \in \text{Strat}_\square} \inf_{\sigma_\bigcirc \in \text{Strat}_\bigcirc} \text{Wt}(\text{Play}(\ell, v, \sigma_\bigcirc, \sigma_\square))$

How to compare them? $\underline{\text{Val}}(\ell, v) \leq \overline{\text{Val}}(\ell, v)$

Theorem: (continued)

- 1BWTGs are determined: $\underline{\text{Val}}(\ell, v) = \overline{\text{Val}}(\ell, v)$
- Synthesis of $\varepsilon$-optimal strategies for player $\square$ in pseudo-polynomial time (and polynomial in case of non-negative weights)
Sketch of proof

1. **Reduce the space of strategies in the 1BWTG**: restrict to uniform strategies w.r.t. timed behaviors

2. **Build a weighted finite games** $G$ based on a refinement of the region abstraction

3. **Study** $G$

4. **Lift results of** $G$ **to the original 1BWTG**
1. Reduce the space of strategies

Intuition: no need for both players to play far from boundaries of regions

\[ x < 1, x := 0, 0 \]

\[ x > 0 \]
\[ x := 0, 0 \]
\[ x \leq 2 \]
\[ x := 0, 0 \]
\[ x \geq 1 \]
\[ x := 0, 0 \]
\[ x \geq 1, 1 \]
\[ x > 1, 1 \]

Regions: \( \{0\}, (0, 1), \{1\}, (1, 2), \{2\}, (2, +\infty) \)

*Player \( \bigcirc \) wants to leave as soon as possible a state with rate \( p^+ \), and wants to stay as long as possible in a state with rate \( p^- \): so, he will always play \( \eta \)-close to a boundary...*

**Lemma:**
Both players can play arbitrarily close to boundaries w.l.o.g., i.e., for every \( \eta \)

\[ \underline{\text{Val}}(\ell, v) \leq \text{Val}(\ell, v) \leq \overline{\text{Val}}(\ell, v) \leq \overline{\text{Val}}^\eta(\ell, v) \]
2. Weighted finite game abstraction

η-regions: \( \{0\}, (0, \eta), (1 - \eta, 1), \{1\}, (1, 1 + \eta), (2 - \eta, 2), \{2\}, (2, +\infty) \)
2. Weighted finite game abstraction
3. Study $\mathcal{G}$: values and optimal strategies

Optimal value: $\text{Val}_\mathcal{G}(\ell_1, \{0\}) = +2$ (for both players)
4. Lift results of $G$ to the original 1BWTG

Reconstruct strategies in the 1BWTG from optimal strategies of $G$

Lemma:
For all $\varepsilon > 0$, there exists $\eta > 0$ such that:

$$\text{Val}_G(\ell, \{0\}) - \varepsilon \leq \text{Val}^\eta(\ell, 0) \leq \text{Val}(\ell, 0) \leq \overline{\text{Val}}(\ell, 0) \leq \overline{\text{Val}}^\eta(\ell, 0) \leq \text{Val}_G(\ell, \{0\}) + \varepsilon$$
4. Lift results of $\mathcal{G}$ to the original 1BWTG

Reconstruct strategies in the 1BWTG from optimal strategies of $\mathcal{G}$

**Lemma:**

For all $\varepsilon > 0$, there exists $\eta > 0$ such that:

$$\text{Val}_{\mathcal{G}}(\ell, \{0\}) - \varepsilon \leq \text{Val}^{n}(\ell, 0) \leq \text{Val}(\ell, 0) \leq \overline{\text{Val}}(\ell, 0) \leq \overline{\text{Val}}^{n}(\ell, 0) \leq \text{Val}_{\mathcal{G}}(\ell, \{0\}) + \varepsilon$$

- So $\text{Val}(\ell, 0) = \overline{\text{Val}}(\ell, 0)$, i.e., determination
- $\varepsilon$-optimal strategies for both players
  - Finite memory for player $\bigcirc$, because finite memory in weighted finite games
  - Infinite memory for player $\square$ (even though memoryless in weighted finite games), because it needs to ensure convergence of its differences between the 1BWTG and $\mathcal{G}$
- Overall complexity: pseudo-polynomial (polynomial if non-negative weights) in the size of $\mathcal{G}$, which is polynomial in the 1BWTG (because 1 clock)
## Summary and Future Work

### Results

- 1BWTGs are determined: $\overline{\text{Val}}(\ell, v) = \overline{\text{Val}}(\ell, v)$
- Computation of the values in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Synthesis of $\varepsilon$-optimal strategies for both players in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Strategy complexity: finite memory for player $\bigcirc$, infinite memory for player $\blacksquare$
- Other results obtained in this context: undecidability results due to the presence of negative weights...
- Implementation and test of this algorithm for real instances
- Extensions to a richer model of priced timed games with negative weights: careful since players may need to play far from boundaries in case of 2 clocks, or 1 clock and 3 distinct rates...
- Consider other objectives, e.g., timed bounded restrictions, leading to decidability in some cases
Summary and Future Work

Results

- 1BWTGs are determined: $\overline{\text{Val}}(\ell, v) = \overline{\text{Val}}(\ell, v)$
- Computation of the values in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Synthesis of $\varepsilon$-optimal strategies for both players in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Strategy complexity: finite memory for player $\bigcirc$, infinite memory for player $\square$

Other results obtained in this context: undecidability results due to the presence of negative weights...
### Summary and Future Work

#### Results

- 1BWTGs are determined: $\text{Val}(\ell,v) = \overline{\text{Val}}(\ell,v)$
- Computation of the values in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Synthesis of $\varepsilon$-optimal strategies for both players in pseudo-polynomial time (and polynomial in case of non-negative weights)
- Strategy complexity: finite memory for player $\bigcirc$, infinite memory for player $\square$

Other results obtained in this context: undecidability results due to the presence of negative weights...

- Implementation and test of this algorithm for real instances
- Extensions to a richer model of priced timed games with negative weights: careful since players may need to play far from boundaries in case of 2 clocks, or 1 clock and 3 distinct rates...
- Consider other objectives, e.g., timed bounded restrictions, leading to decidability in some cases


