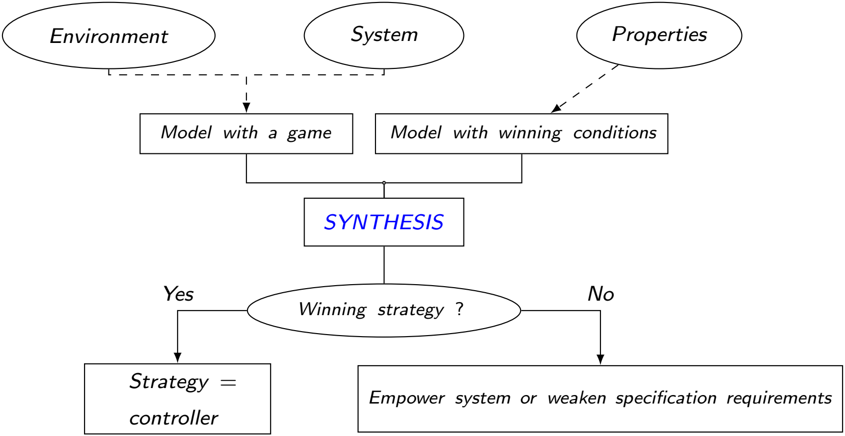
On the complexity of heterogeneous multidimensional quantitative games

Véronique Bruyère[1] Quentin Hautem[1] Jean-François Raskin[2]

University of Mons[1], Université Libre of Bruxelles[2]

August 23, 2016. CONCUR 2016

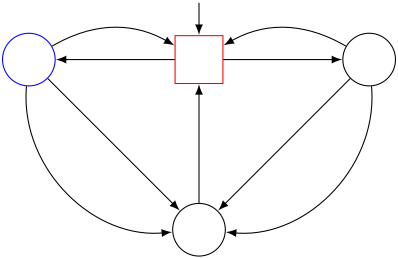
Synthesis via Game Theory



Model

Zero-sum games played on finite graph:

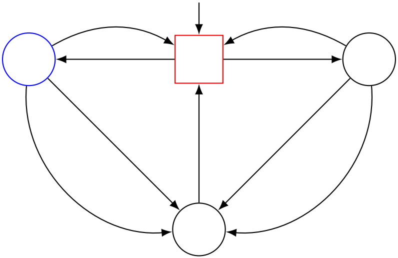
- System vs. Environment : antagonistic
- Turn-based games



Model

Zero-sum games played on finite graph:

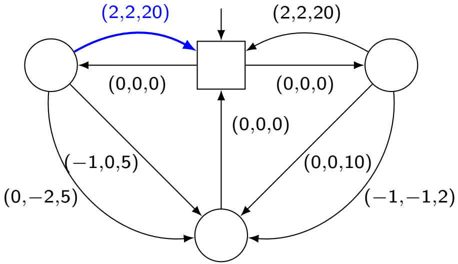
- System vs. Environment : antagonistic
- Turn-based games



A strategy is a function mapping each history of the game to a successor
A play is winning for player 1 if it satisfies its winning condition (called objective)

Model

Multi-dimensional weighted zero-sum games played on finite graph:
- Weight for energy, time consumption, . . .



Known results

- Uni-dimensional ([Jur98] [CDRR15])

	Inf	LimInf	Sup	LimSup	MP	En.	WMP
Complexity	P-complete				$NP \cap coNP$		P-c
P1 memory	memoryless						exponential
P2 memory	memoryless						exponential

- Multi-dimensional: Homogeneous intersection ([CDHR10] [CDRR15])

	En	$\underline{\text{MP}}$	$\overline{\text{MP}}$	WMP
Complexity	coNP-c		$NP \cap coNP$	EXPTIME-c
P1 memory	finite-memory	infinite-memory		exponential
P2 memory	memoryless			exponential

- Boolean combinations of $\overline{\text{MP}}$ and $\underline{\text{MP}}$: undecidable. [Vel15]

Problem

We consider here heterogeneous objectives.

- One objective by dimension
- Objective : measure of the play $\sim$ threshold ν with $\sim \in \{\geq, \leq, >, <\}$[1]

The *threshold problem* asks to decide whether player 1 has a winning strategy for Ω from an initial vertex v_0.

[1] W.l.o.g we can assume $\sim = \geq$ and $\nu = 0$

Quantitative measures studied[2]:

- Inf (Sup) : minimum (maximum) weight seen
- LimInf (LimSup): minimum (maximum) weight infinitely often seen
- *WindowMeanPayoff* (WMP): average weight over a local window sliding along the play

[2] All those measures are ω-regular

Quantitative measures studied:

- Inf (Sup) : minimum (maximum) weight seen
- LimInf (LimSup): minimum (maximum) weight infinitely often seen
- *WindowMeanPayoff* (WMP): average weight over a local window sliding along the play

Example: design a system

- ϕ_1 : with a good window mean-response time (WMP),
- ϕ_2 : that avoids too slow reaction (Inf) and
- ϕ_3 : that does not exceed some peak energy consumption in the long run (LimSup).

$$\rightsquigarrow \phi_1 \wedge \phi_2 \wedge \phi_3$$

Quantitative measures studied:

- Inf (Sup) : minimum (maximum) weight seen
- LimInf (LimSup): minimum (maximum) weight infinitely often seen
- *WindowMeanPayoff* (WMP): average weight over a local window sliding along the play

Example:

- hypothesis (ψ) : the frequency of requests from the environment is below some threshold (expressible as a WMP)

$$\rightsquigarrow \psi \rightarrow (\phi_1 \wedge \phi_2 \wedge \phi_3)$$

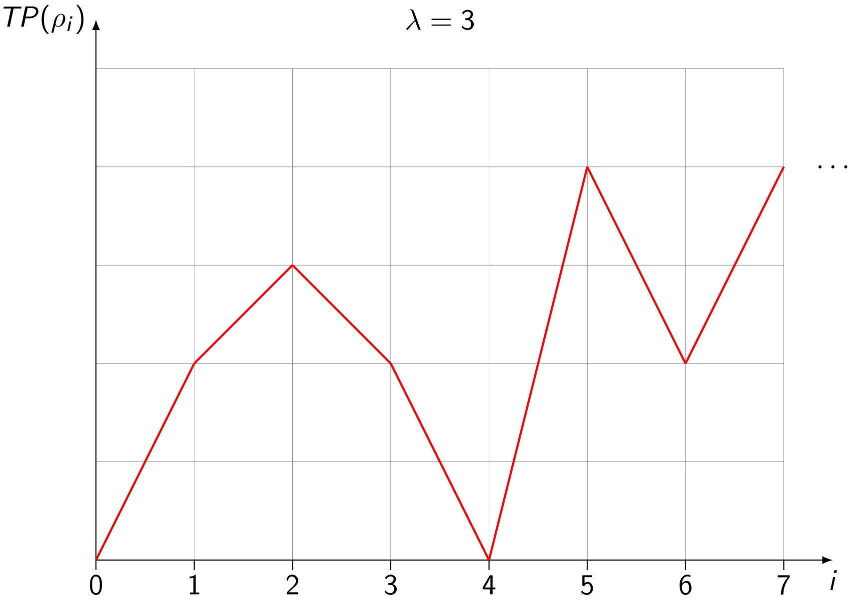
Quantitative measures studied:

- Inf (Sup) : minimum (maximum) weight seen
- LimInf (LimSup): minimum (maximum) weight infinitely often seen
- *WindowMeanPayoff* (WMP): average weight over a local window sliding along the play

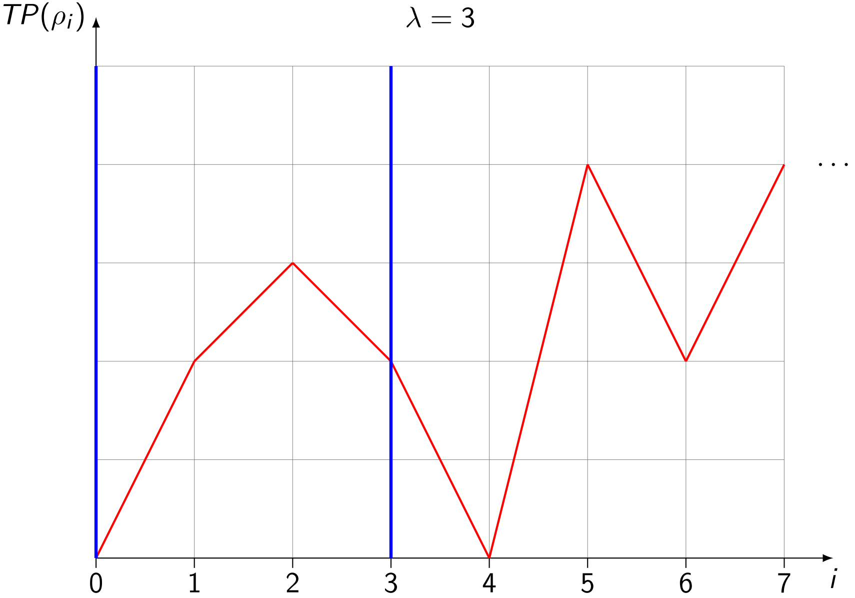
Definition

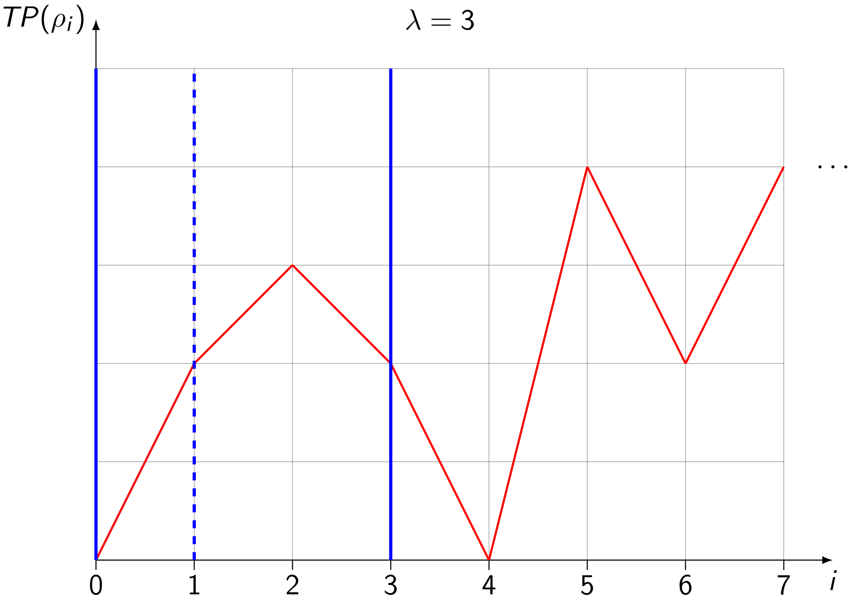
Given a threshold $\nu \in \mathbb{Q}$ and a window size $\lambda \in \mathbb{N} \setminus \{0\}$,

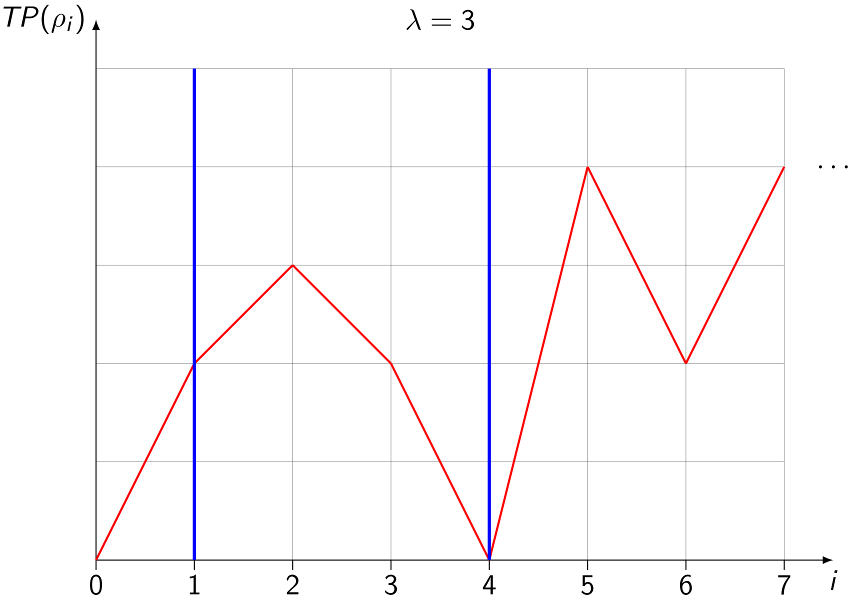
$$\mathrm{WMP}(\lambda, \nu) = \{\rho \in \mathrm{Plays}(G) \mid \forall k \geq 0, \exists l \in \{1, \ldots, \lambda\}, \mathrm{MP}(\rho_{[k,k+l]}) \geq \nu\}.$$

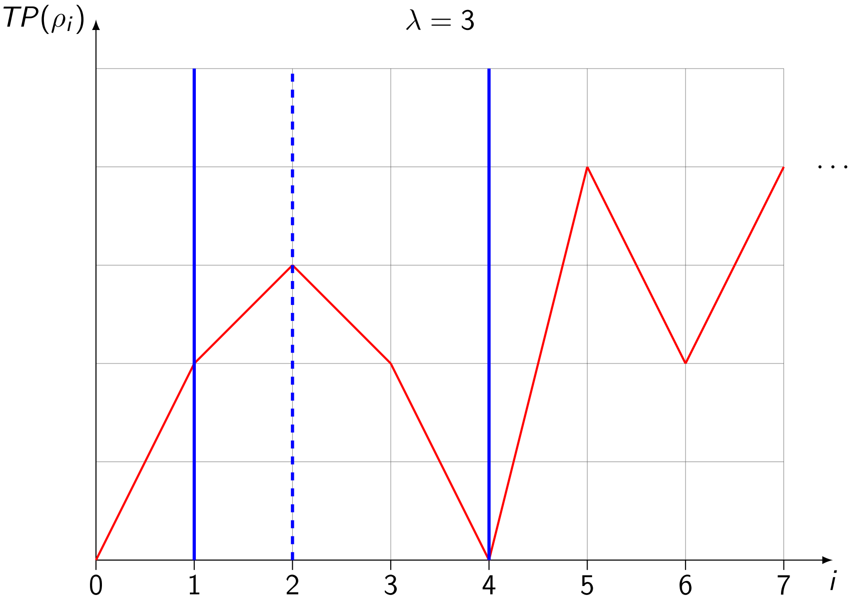

$TP(\rho_i)$
$\lambda = 3$
0
1
2
3
4
5
6
7
i
. . .

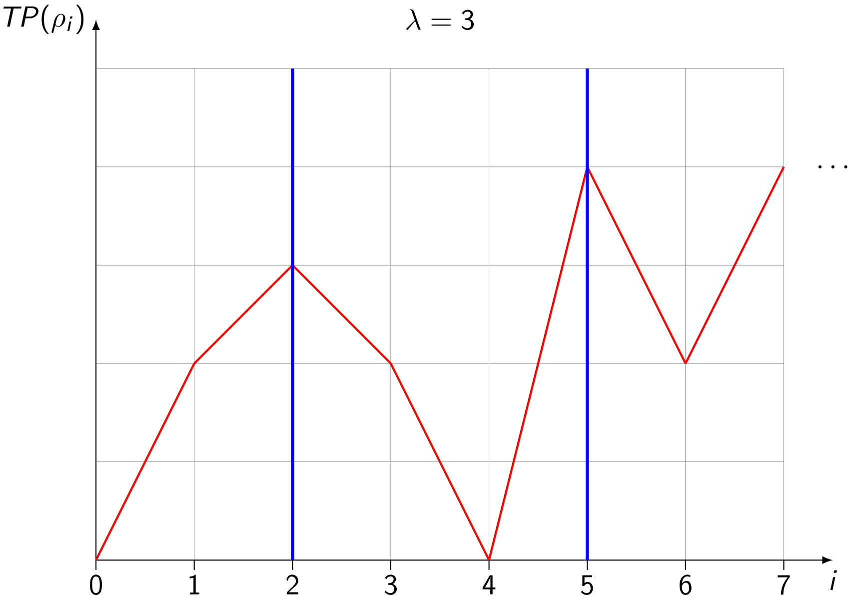
$\lambda = 3$

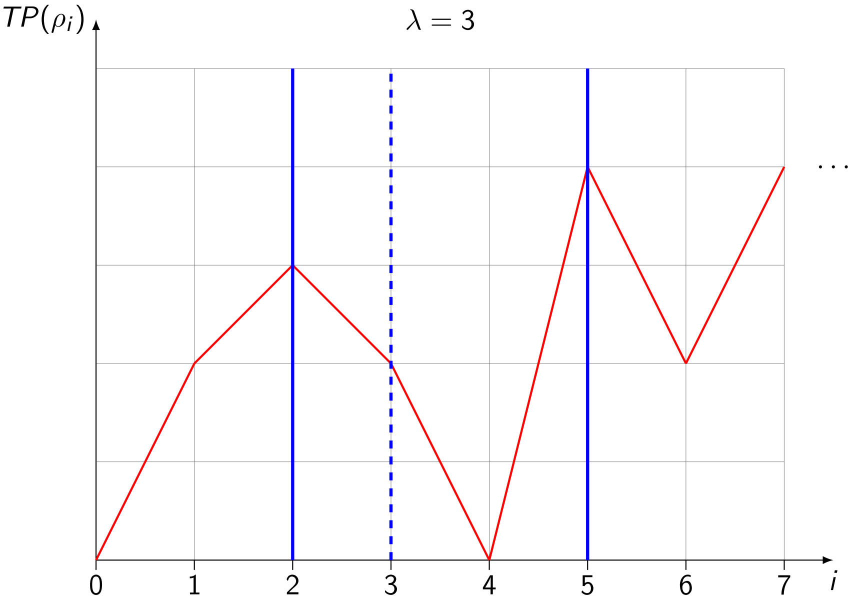


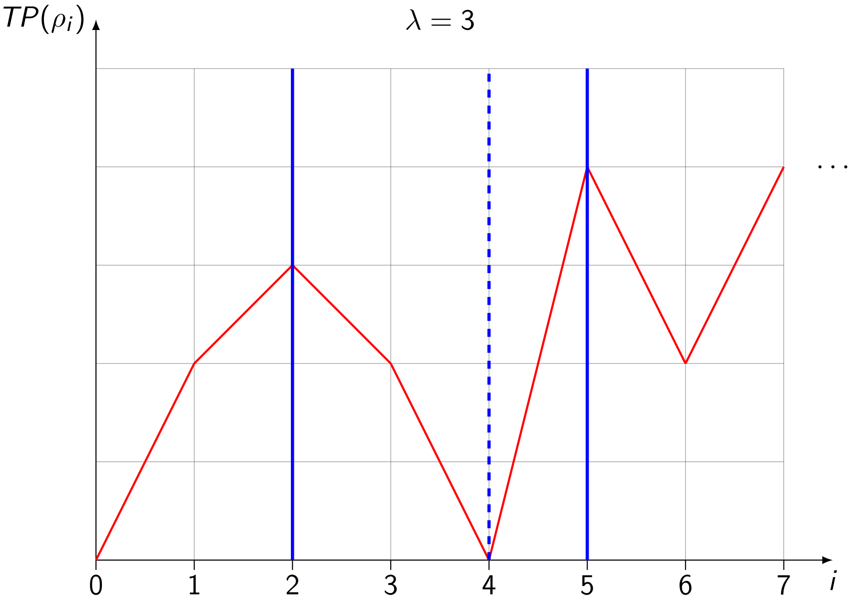

$TP(\rho_i)$
$\lambda = 3$
...
0
1
2
3
4
5
6
7
i

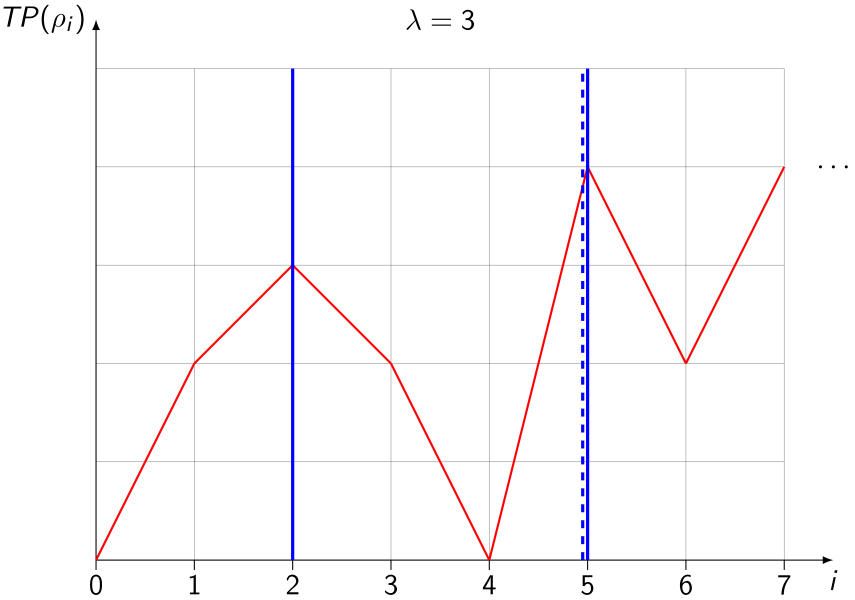

$TP(\rho_i)$
$\lambda = 3$
. . .
0
1
2
3
4
5
6
7
i


$TP(\rho_i)$
$\lambda = 3$
...
0
1
2
3
4
5
6
7
i


$TP(\rho_i)$
$\lambda = 3$
. . .
0 1 2 3 4 5 6 7
i


$TP(\rho_i)$
$\lambda = 3$
0
1
2
3
4
5
6
7
i
...


$TP(\rho_i)$
$\lambda = 3$
. . .
0
1
2
3
4
5
6
7
i


$TP(\rho_i)$
$\lambda = 3$
...
0
1
2
3
4
5
6
7
i

WMP(λ, 0)

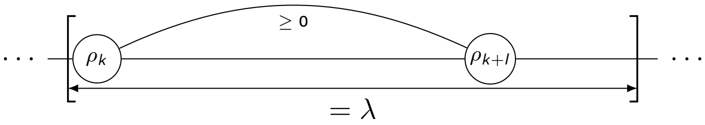
Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,

WMP(λ, 0)

Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,



WMP$(\lambda, 0)$

Window at position k is

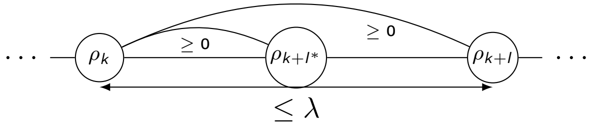
- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,

- first-closed in $k + l^*$ if l^* is minimal,

WMP$(\lambda, 0)$

Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,
- first-closed in $k + l^*$ if l^* is minimal,



WMP$(\lambda, 0)$

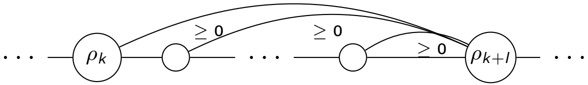
Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,
- first-closed in $k + l^*$ if l^* is minimal,
- inductively-closed in $k + l$ if it closed in $k + l$ and this is also the case for each $k' \in \{k + 1, \ldots, k + l\}$.

WMP(λ, 0)

Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,
- first-closed in $k + l^*$ if l^* is minimal,
- inductively-closed in $k + l$ if it closed in $k + l$ and this is also the case for each $k' \in \{k + 1, \ldots, k + l\}$.



WMP(λ, 0)

Window at position k is

- closed in $k + l$ if $\exists l \in \{1, \ldots, \lambda\}$ s.t. $TP(\rho_{[k,k+l]}) \geq 0$,
- first-closed in $k + l^*$ if l^* is minimal,
- inductively-closed in $k + l$ if it closed in $k + l$ and this is also the case for each $k' \in \{k + 1, \ldots, k + l\}$.

A window that is first-closed is inductively-closed.

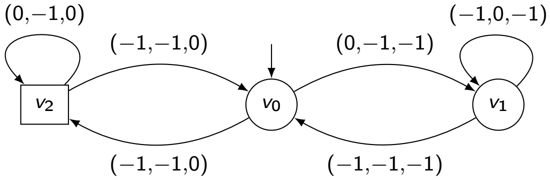
Questions

Threshold problem for $\Omega = \bigcap_{m=1}^{n} \Omega_m$ with $\Omega_m \in \{$WMP, Inf, Sup, LimInf, LimSup$\}$.

- Is the threshold problem decidable ?
- If yes, what is the complexity class ?
- How much memory is needed for winning strategies ?

Example

$\Omega = \text{LimSup}(0) \cap \text{Sup}(0) \cap \text{LimSup}(0)$



σ_1: Loop on v_1 then switch between v_1 and v_2

Results

<table>
<tr><th>Objectives</th><th>Complexity class</th><th>Player 1 memory</th><th>Player 2 memory</th></tr>
<tr><td>(CNF/DNF) Boolean combination of $\underline{\text{MP}}$, $\overline{\text{MP}}$ [Vel15]</td><td>Undecidable</td><td>infinite</td><td>infinite</td></tr>
<tr><td>(CNF/DNF) Boolean combinaison of
WMP, Inf, Sup, LimInf, LimSup</td><td rowspan="3">EXPTIME-complete</td><td rowspan="4" colspan="2">exponential</td></tr>
<tr><td>Intersection of WMP, Inf, Sup, LimInf, LimSup</td></tr>
<tr><td>Intersection of WMP [CDRR15]</td></tr>
<tr><td rowspan="2">Intersection of Inf, Sup, LimInf, LimSup
and refinements</td><td>PSPACE-complete</td></tr>
<tr><td colspan="3">See Table of Section "PSPACE fragment"</td></tr>
<tr><td>Intersection of $\underline{\text{MP}}$ [VCD+15]</td><td>coNP-complete</td><td rowspan="2">infinite</td><td rowspan="3">memoryless</td></tr>
<tr><td>Intersection of $\overline{\text{MP}}$ [VCD+15]</td><td rowspan="2">NP $\cap$ coNP</td></tr>
<tr><td>Unidimensional MP [ZP96, BCD+11]</td><td>memoryless</td></tr>
<tr><td>Unidimensional WMP [CDRR15]
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$</td><td>P-complete
(Polynomial windows)</td><td colspan="2">pseudo-polynomial</td></tr>
<tr><td>Unidimensional Inf, Sup, LimInf, LimSup [GTW02]</td><td>P-complete</td><td colspan="2">memoryless</td></tr>
</table>

Results

<table>
<tr><th>Objectives</th><th>Complexity class</th><th>Player 1 memory</th><th>Player 2 memory</th></tr>
<tr><td>(CNF/DNF) Boolean combinaison of
WMP, Inf, Sup, LimInf, LimSup</td><td rowspan="2">EXPTIME-complete</td><td colspan="2" rowspan="3">exponential</td></tr>
<tr><td>Intersection of WMP, Inf, Sup, LimInf, LimSup</td></tr>
<tr><td rowspan="2">Intersection of Inf, Sup, LimInf, LimSup
and refinements</td><td>PSPACE-complete</td></tr>
<tr><td colspan="3">See Table of Section "Inf, Sup, LimInf, LimSup"</td></tr>
<tr><td>$\text{WMP} \cap \cap \Omega_m$ with $\Omega_m \in \{\text{Inf, Sup, LimInf, LimSup}\}$</td><td>P-complete
(Polynomial windows)</td><td colspan="2">pseudo-polynomial</td></tr>
</table>

General result

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	See Table of Section "Inf, Sup, LimInf, LimSup"		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$	P-complete (Polynomial windows)	pseudo-polynomial	

General result

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	See Table of Section "Inf, Sup, LimInf, LimSup"		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$	P-complete (Polynomial windows)	pseudo-polynomial	

- Intersection

Membership : use the exponential reduction inspired from [CDRR15] and solve a generalized-Büchi $\cap$ co-Büchi game.
Hardness: EXPTIME-hard even for two WMP objectives [CDRR15].

General result

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	See Table of Section "Inf, Sup, LimInf, LimSup"		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in$ {Inf, Sup, LimInf, LimSup}	P-complete (Polynomial windows)	pseudo-polynomial	

- CNF/DNF Boolean combination

Proof : Use the same reduction as before and solve a Rabin game with d pairs.

Remark: Undecidable even for Boolean combination of $\overline{\text{MP}}$ and $\underline{\text{MP}}$ [Vel15].

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	See next Table		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$	P-complete (Polynomial windows)	pseudo-polynomial	

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	See next Table		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$	P-complete (Polynomial windows)	pseudo-polynomial	

Proof: Use a polynomial reduction to obtain a game (G', Ω') with

$$\Omega' = \text{GenReach}(U_1, \dots, U_{j-1}) \cap \text{GenBuchi}(U_j, \dots, U_{i-1}) \cap \text{CoBuchi}(U_i)^2.$$

Solve the generalized-Büchi $\cap$ co-Büchi game and then the generalized-reachability game.

[2] We have transformed safety objectives to co-Büchi objectives

Corollary

Inf	Sup	LimInf	LimSup	Complexity	player 1 memory	player 2 memory
any	any	any	any	PSPACE-c	finite-memory	finite-memory
any	≤ 1	any	any	P-complete	finite-memory	memoryless
any	0	any	≤ 1	P-complete	memoryless	memoryless
any	1	0	0	P-complete	memoryless	memoryless

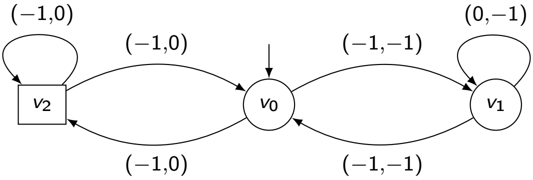
Corollary

Inf	Sup	LimInf	LimSup	Complexity	player 1 memory	player 2 memory
any	any	any	any	PSPACE-c	finite-memory	finite-memory
any	≤ 1	any	any	P-complete	finite-memory	memoryless
any	0	any	≤ 1	P-complete	memoryless	memoryless
any	1	0	0	P-complete	memoryless	memoryless

- Note the polynomial fragment
- Complete analysis

Example

$\Omega = \text{Sup}(0) \cap \text{LimSup}(0)$



Player 1 needs memory.

<table>
<tr><th>Objectives</th><th>Complexity class</th><th>Player 1 memory</th><th>Player 2 memory</th></tr>
<tr><td>(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup</td><td rowspan="2">EXPTIME-complete</td><td colspan="2" rowspan="3">exponential</td></tr>
<tr><td>Intersection of WMP, Inf, Sup, LimInf, LimSup</td></tr>
<tr><td rowspan="2">Intersection of Inf, Sup, LimInf, LimSup and refinements</td><td>PSPACE-complete</td></tr>
<tr><td colspan="3">Previous Table</td></tr>
<tr><td>WMP $\cap \cap \Omega_m$ with $\Omega_m \in$ {Inf, Sup, LimInf, LimSup}</td><td>P-complete (Polynomial windows)</td><td colspan="2">pseudo-polynomial</td></tr>
</table>

Objectives	Complexity class	Player 1 memory	Player 2 memory
(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup	EXPTIME-complete	exponential	
Intersection of WMP, Inf, Sup, LimInf, LimSup			
Intersection of Inf, Sup, LimInf, LimSup and refinements	PSPACE-complete		
	Previous Table		
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$	P-complete (Polynomial windows)	pseudo-polynomial	

- Two WMP objectives lead to EXPTIME-hardness,
- Several Sup objectives lead to PSPACE-hardness,
- Same kind of objectives in the intersection (n fixed for Sup).

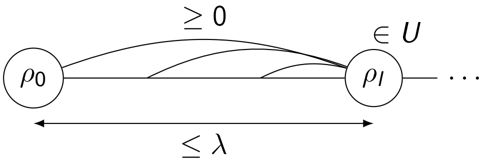
WMP ∩ Sup

Reduction: $(G', \Omega'_1 \cap \Omega'_2)$ with $\Omega'_1 = \text{WMP}$ and $\Omega'_2 = \text{Reach}$.

Need genuine new tools to deal with windows and the qualitative objectives.

First tool:

$\text{ICWEnd}^{\lambda}(U)$:



$\mathsf{ICWEnd}^{\lambda}(U)$

Algorithm 1 ICWEnd

Require: 1-weighted game structure $G = (V_1, V_2, E, w)$, set $U \subseteq V$, window size $\lambda \in \mathbb{N} \setminus \{0\}$

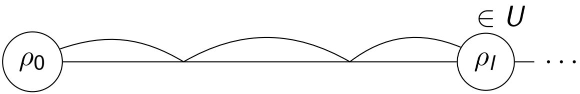
Ensure: $\mathsf{Win}_1^{\mathsf{ICWEnd}^{\lambda}(U)}$

for all $v \in V$ **do**
    if $v \in U$ **then**
        $C_0(v) \leftarrow 0$
    else
        $C_0(v) \leftarrow -\infty$
for all $l \in \{1, \dots, \lambda\}$ **do**
    for all $v \in V_1$ **do**
        $C_l(v) \leftarrow \max_{(v,v') \in E}\{w(v,v') \oplus \max\{C_0(v'), C_{l-1}(v')\}\}$
    for all $v \in V_2$ **do**
        $C_l(v) \leftarrow \min_{(v,v') \in E}\{w(v,v') \oplus \max\{C_0(v'), C_{l-1}(v')\}\}$
return $\{v \in V \mid C_{\lambda}(v) \geq 0\}$

$\mathsf{GDEnd}^{\lambda}(U)$

Second tool: Generalization of the p-attractor of a set U while dealing with windows.

$\mathsf{GDEnd}^{\lambda}(U)$:



Require: 1-weighted game structure $G = (V_1, V_2, E, w)$, subset $U \subseteq V$, window size $\lambda \in \mathbb{N} \setminus \{0\}$

Ensure: $\mathsf{Win}_1^{\mathsf{GDEnd}^{\lambda}(U)}(G)$

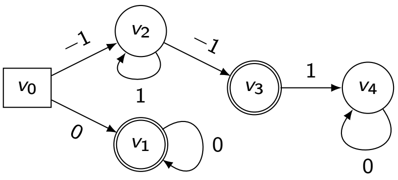
```
k ← 0
X_0 ← U
repeat
    X_{k+1} ← X_k ∪ ICWEnd(G, X_k, λ)
    k ← k + 1
until X_k = X_{k−1}
return X_k
```

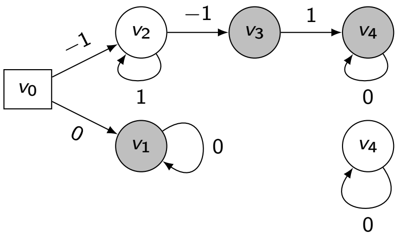
WMP ∩ Reach

- Use algorithm $\mathsf{GDEnd}^{\lambda}(U')$ on a modified graph.
  U' is the set of vertices that denote that we have visited U and that are winning for the WMP objective.

WMP ∩ Reach

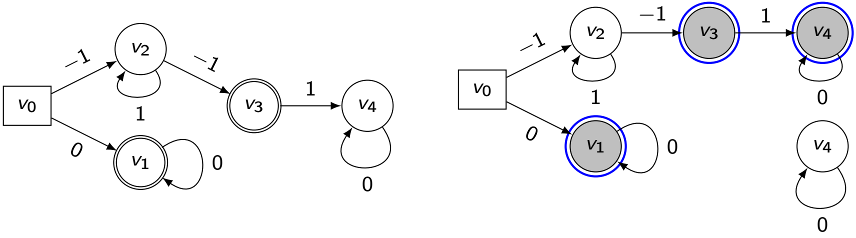
- Use algorithm $\mathsf{GDEnd}^{\lambda}(U')$ on a modified graph. U' is the set of vertices that denote that we have visited U and that are winning for the WMP objective.
- Example ($\lambda = 2$):





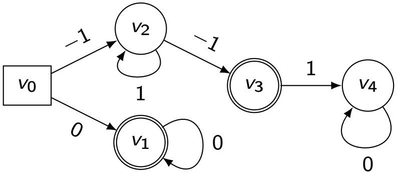
WMP ∩ Reach

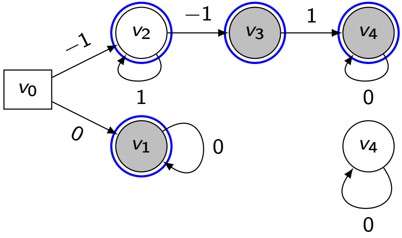
- Use algorithm $\mathrm{GDEnd}^{\lambda}(U')$ on a modified graph.
  U' is the set of vertices that denote that we have visited U and that are winning for the WMP objective.
- Example ($\lambda = 2$):



WMP ∩ Reach

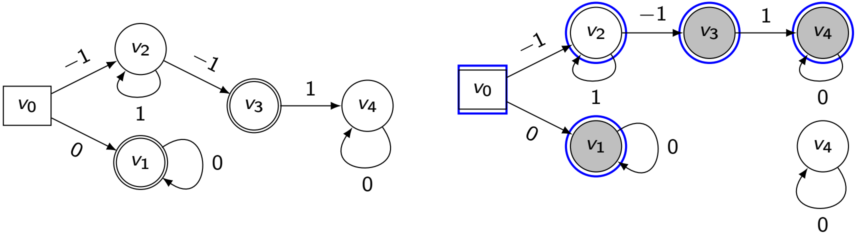
- Use algorithm $\mathsf{GDEnd}^{\lambda}(U')$ on a modified graph.
  U' is the set of vertices that denote that we have visited U and that are winning for the WMP objective.
- Example ($\lambda = 2$):





WMP ∩ Reach

- Use algorithm $\text{GDEnd}^{\lambda}(U')$ on a modified graph. U' is the set of vertices that denote that we have visited U and that are winning for the WMP objective.
- Example ($\lambda = 2$):



Overview

<table>
<tr><th>Objectives</th><th>Complexity class</th><th>Player 1 memory</th><th>Player 2 memory</th></tr>
<tr><td>(CNF/DNF) Boolean combination of $\underline{\text{MP}}$, $\overline{\text{MP}}$ [Vel15]</td><td>Undecidable</td><td>infinite</td><td>infinite</td></tr>
<tr><td>(CNF/DNF) Boolean combinaison of WMP, Inf, Sup, LimInf, LimSup</td><td rowspan="3">EXPTIME-complete</td><td colspan="2" rowspan="4">exponential</td></tr>
<tr><td>Intersection of WMP, Inf, Sup, LimInf, LimSup</td></tr>
<tr><td>Intersection of WMP [CDRR15]</td></tr>
<tr><td rowspan="2">Intersection of Inf, Sup, LimInf, LimSup and refinements</td><td>PSPACE-complete</td></tr>
<tr><td colspan="3">See Table of Section "Inf, Sup, LimInf, LimSup"</td></tr>
<tr><td>Intersection of $\underline{\text{MP}}$ [VCD+15]</td><td>coNP-complete</td><td rowspan="2">infinite</td><td rowspan="3">memoryless</td></tr>
<tr><td>Intersection of $\overline{\text{MP}}$ [VCD+15]</td><td rowspan="2">NP $\cap$ coNP</td></tr>
<tr><td>Unidimensional MP [ZP96, BCD+11]</td><td>memoryless</td></tr>
<tr><td>Unidimensional WMP [CDRR15]
WMP $\cap \cap \Omega_m$ with $\Omega_m \in \{$Inf, Sup, LimInf, LimSup$\}$</td><td>P-complete
(Polynomial windows)</td><td colspan="2">pseudo-polynomial</td></tr>
<tr><td>Unidimensional Inf, Sup, LimInf, LimSup [GTW02]</td><td>P-complete</td><td colspan="2">memoryless</td></tr>
</table>

Future Work

- General results for ω-regular objectives,
- General results for multidimensional lexicographic games,
- Mix non ω-regular objectives,

- Value of WMP.

Thank you for your attention

Lubos Brim, Jakub Chaloupka, Laurent Doyen, Raffaella Gentilini, and Jean-François Raskin.
Faster algorithms for mean-payoff games.
Formal Methods in System Design, 38(2):97–118, 2011.

Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, and Jean-François Raskin.
Generalized mean-payoff and energy games.
In Kamal Lodaya and Meena Mahajan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2010, December 15-18, 2010, Chennai, India*, volume 8 of *LIPIcs*, pages 505–516. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2010.

Krishnendu Chatterjee, Laurent Doyen, Mickael Randour, and Jean-François Raskin.
Looking at mean-payoff and total-payoff through windows.
Inf. Comput., 242:25–52, 2015.

Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors.
Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001], volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.

Marcin Jurdzinski.
Deciding the winner in parity games is in UP n co-up.
Inf. Process. Lett., 68(3):119–124, 1998.

Yaron Velner, Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, Alexander Moshe Rabinovich, and Jean-François Raskin.
The complexity of multi-mean-payoff and multi-energy games.
Inf. Comput., 241:177–196, 2015.

Yaron Velner.
Robust multidimensional mean-payoff games are undecidable.
In Andrew M. Pitts, editor, *Foundations of Software Science and Computation Structures - 18th International Conference, FoSSaCS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings*, volume 9034 of *Lecture Notes in Computer Science*, pages 312–327. Springer, 2015.

Uri Zwick and Mike Paterson.
The complexity of mean payoff games on graphs.
Theor. Comput. Sci., 158(1&2):343–359, 1996.