Synthesis in Multi-Criteria Quantitative Games

Mickael Randour

Advisors: Véronique Bruyère & Jean-François Raskin

Mons - 18.04.2014

Private PhD Thesis Defense

UMONS Université de Mons

ULB Université Libre de Bruxelles

General context

- Verification and synthesis:
  - ▷ a reactive **system** to *control*,
  - ▷ an *interacting* **environment**,
  - ▷ a **specification** to *enforce*.

General context

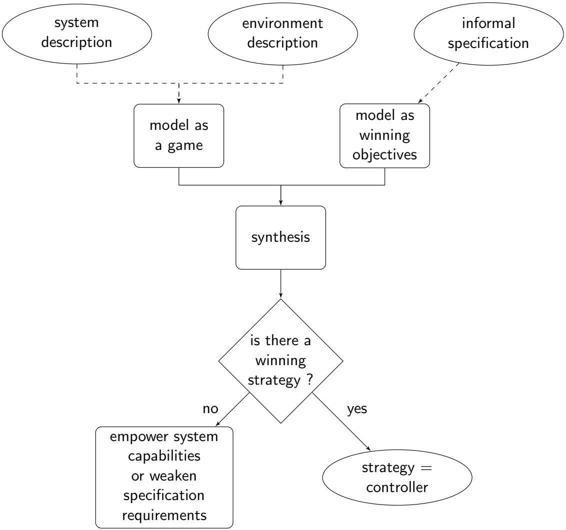
- Verification and synthesis:
  - ▷ a reactive **system** to *control*,
  - ▷ an *interacting* **environment**,
  - ▷ a **specification** to *enforce*.

- *Qualitative* and *quantitative* specifications.

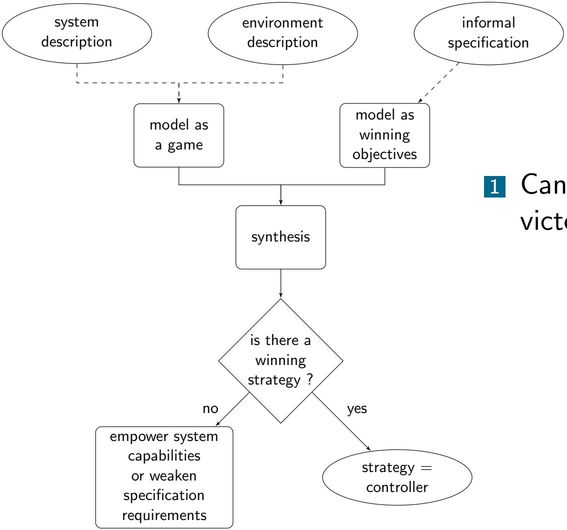
General context

- Verification and synthesis:
  - a reactive **system** to *control*,
  - an *interacting* **environment**,
  - a **specification** to *enforce*.
- *Qualitative* and *quantitative* specifications.
- Focus on **multi-criteria quantitative models**
  - to reason about *trade-offs* and *interplays*.

Synthesis via two-player graph games

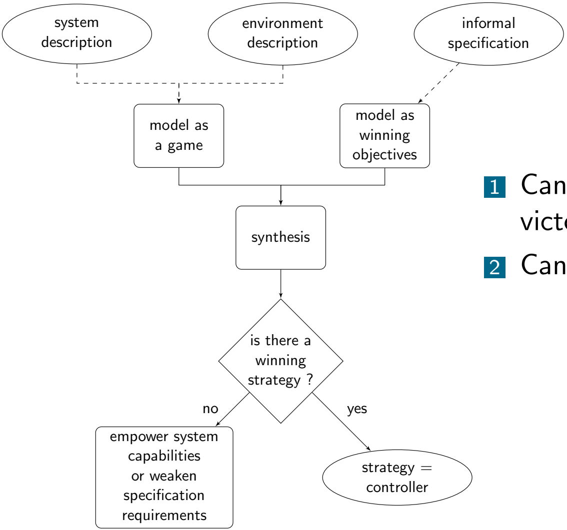


Synthesis via two-player graph games



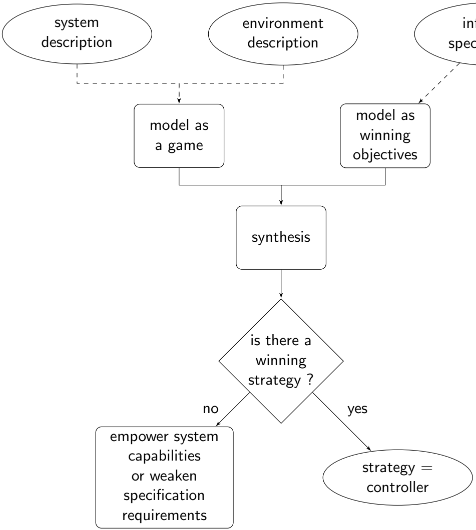
1. Can one player guarantee victory?

Synthesis via two-player graph games



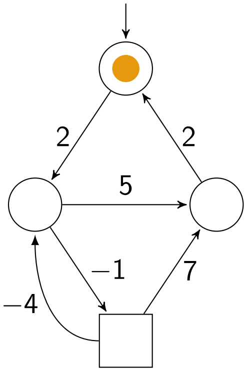
1. Can one player guarantee victory?
2. Can we decide which one?

Synthesis via two-player graph games



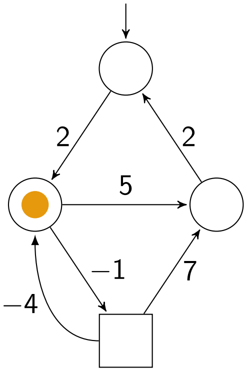
1. Can one player guarantee victory?
2. Can we decide which one?
3. How complex his strategy needs to be?

Quantitative games on graphs



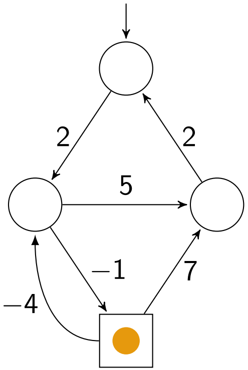
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i\colon \text{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

Quantitative games on graphs



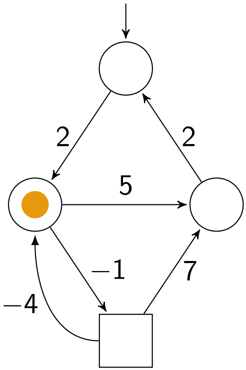
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \rightarrow \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \rightarrow \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i\colon \text{Prefs}_i(G) \rightarrow \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

Quantitative games on graphs



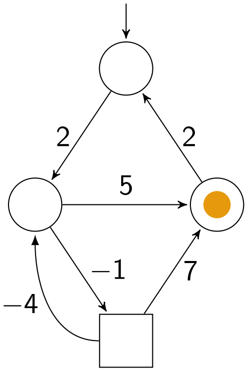
- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i \colon \text{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

Quantitative games on graphs



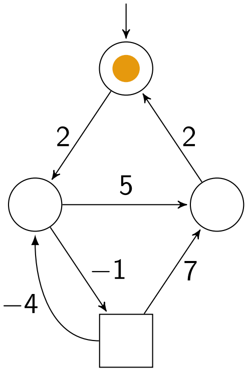
- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i \colon \text{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

Quantitative games on graphs



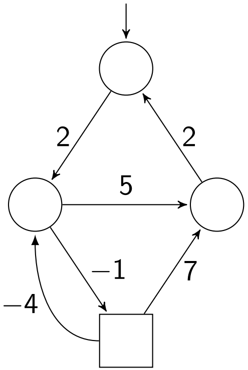
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i\colon \text{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

Quantitative games on graphs



- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i \colon \text{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_\text{u}, \alpha_\text{n})$

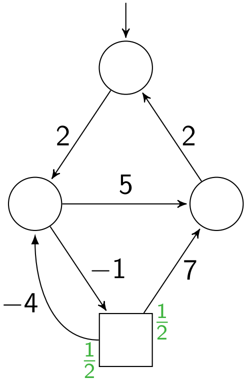
Quantitative games on graphs



Then, $(2, 5, 2)^\omega$

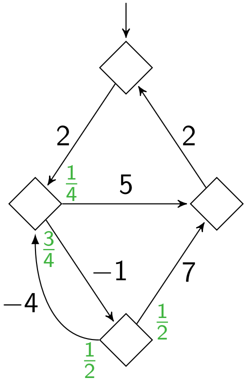
- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \mathrm{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- Players follow *strategies*
  - ▷ $\lambda_i \colon \mathrm{Prefs}_i(G) \to \mathcal{D}(S)$
  - ▷ Finite memory ⇒ stochastic output Moore machine $\mathcal{M}(\lambda_i) = (\mathrm{Mem}, \mathrm{m_0}, \alpha_\mathrm{u}, \alpha_\mathrm{n})$

Markov decision processes



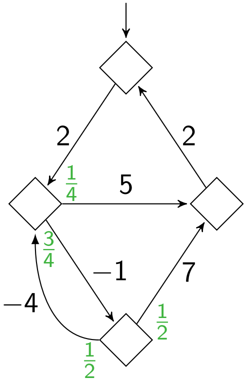
- MDP $P = (\mathcal{G}, S_1, S_\Delta, \Delta)$ with $\Delta \colon S_\Delta \to \mathcal{D}(S)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ stochastic states = □
- MDP = game + strategy of $\mathcal{P}_2$
  - ▷ $P = G[\lambda_2]$

Markov chains



- MC $M = (\mathcal{G}, \delta)$ with $\delta \colon S \to \mathcal{D}(S)$
- MC = MDP + strategy of $\mathcal{P}_1$
  = game + both strategies
  - ▷ $M = P[\lambda_1] = G[\lambda_1, \lambda_2]$

Markov chains



- MC $M = (\mathcal{G}, \delta)$ with $\delta \colon S \to \mathcal{D}(S)$
- MC = MDP + strategy of $\mathcal{P}_1$
  = game + both strategies
  - ▷ $M = P[\lambda_1] = G[\lambda_1, \lambda_2]$
- Event $\mathcal{A} \subseteq \mathsf{Plays}(\mathcal{G})$
  - ▷ probability $\mathbb{P}^{M}_{s_{\mathsf{init}}}(\mathcal{A})$
- Measurable $f \colon \mathsf{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
  - ▷ *expected value* $\mathbb{E}^{M}_{s_{\mathsf{init}}}(f)$

Winning semantics and decision problems

- **Qualitative** objectives - $\phi \subseteq \mathsf{Plays}(G)$
  - ▷ λ_1 *surely winning*: $\forall\, \lambda_2 \in \Lambda_2$, $\mathsf{Outs}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2) \subseteq \phi$
  - ▷ λ_1 *almost-surely winning*: $\forall\, \lambda_2 \in \Lambda_2$, $\mathbb{P}^{G[\lambda_1, \lambda_2]}_{s_{\mathsf{init}}}(\phi) = 1$

Winning semantics and decision problems

- **Qualitative** objectives - $\phi \subseteq \mathsf{Plays}(G)$
  - ▷ λ_1 *surely winning*: $\forall\, \lambda_2 \in \Lambda_2$, $\mathsf{Outs}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2) \subseteq \phi$
  - ▷ λ_1 *almost-surely winning*: $\forall\, \lambda_2 \in \Lambda_2$, $\mathbb{P}^{G[\lambda_1, \lambda_2]}_{s_{\mathsf{init}}}(\phi) = 1$
- **Quantitative** objectives - $f\colon \mathsf{Plays}(G) \rightarrow \mathbb{R} \cup \{-\infty, \infty\}$
  - ▷ *worst-case threshold problem*, $\mu \in \mathbb{Q}$:

    $$\exists?\, \lambda_1 \in \Lambda_1, \forall\, \lambda_2 \in \Lambda_2, \forall\, \pi \in \mathsf{Outs}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2),\ f(\pi) \geq \mu$$
  - ▷ *expected value threshold problem* (MDP), $\nu \in \mathbb{Q}$:

    $$\exists?\, \lambda_1 \in \Lambda_1, \mathbb{E}^{P[\lambda_1]}_{s_{\mathsf{init}}}(f) \geq \nu$$

Classical qualitative objectives

- $\text{Reach}_G(T) = \{\pi = s_0 s_1 s_2 \ldots \in \text{Plays}(G) \mid \exists\, i \in \mathbb{N},\ s_i \in T\}$
- $\text{Buchi}_G(T) = \{\pi = s_0 s_1 s_2 \ldots \in \text{Plays}(G) \mid \text{Inf}(\pi) \cap T \neq \emptyset\}$
- $\text{Parity}_G = \{\pi = s_0 s_1 s_2 \ldots \in \text{Plays}(G) \mid \text{Par}(\pi) \bmod 2 = 0\}$

Classical quantitative objectives and value functions

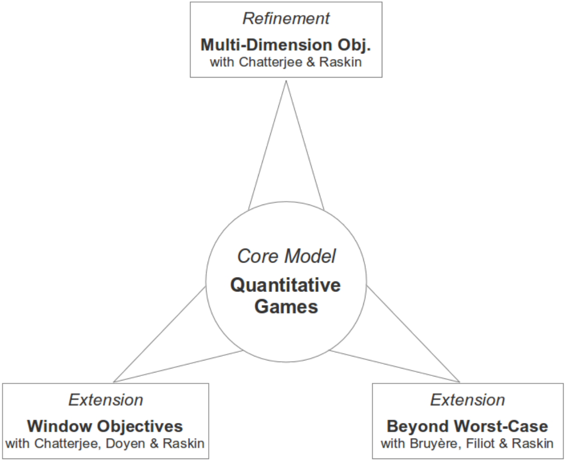
- *Total-payoff*: $\underline{\text{TP}}(\pi) = \liminf_{n \to \infty} \sum_{i=0}^{i=n-1} w((s_i, s_{i+1}))$
- *Mean-payoff*: $\underline{\text{MP}}(\pi) = \liminf_{n \to \infty} \frac{1}{n} \sum_{i=0}^{i=n-1} w((s_i, s_{i+1}))$
- *Shortest path*: truncated sum up to first visit of $T \subseteq S$
- *Energy*: keep the running sum positive at all times

Single-criterion models - known results

		reachability	Büchi	parity
GAMES *sure sem.*	complexity	P-c.		UP $\cap$ coUP
	$\mathcal{P}_1$ mem.	pure memoryless		
	$\mathcal{P}_2$ mem.			
MDPS *almost-sure sem.*	complexity	P-c.		
	$\mathcal{P}_1$ mem.	pure memoryless		

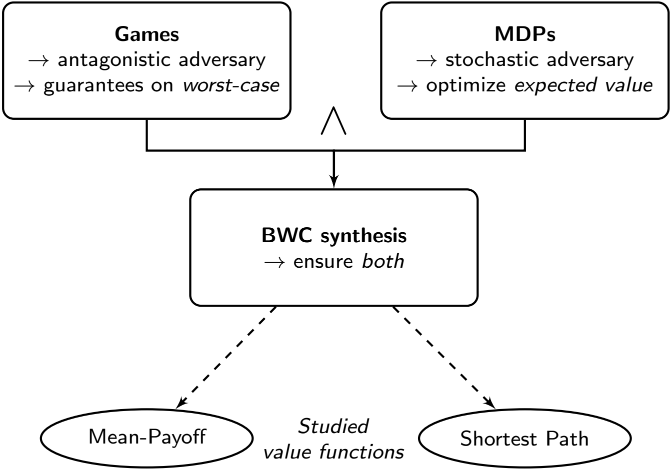
		TP	MP	SP	EG
GAMES *worst-case*	complexity	UP $\cap$ coUP		P-c.	UP $\cap$ coUP
	$\mathcal{P}_1$ mem.	pure memoryless			
	$\mathcal{P}_2$ mem.				
MDPS *expected value*	complexity	P-c.			n/a
	$\mathcal{P}_1$ mem.	pure memoryless			

Contributions

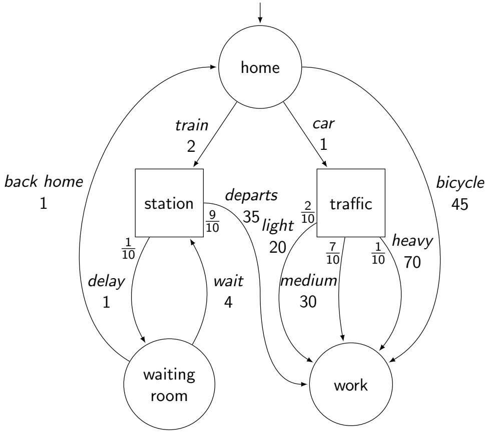


Shift from single-criterion models to **multi-criteria** ones.

Combining two classical models

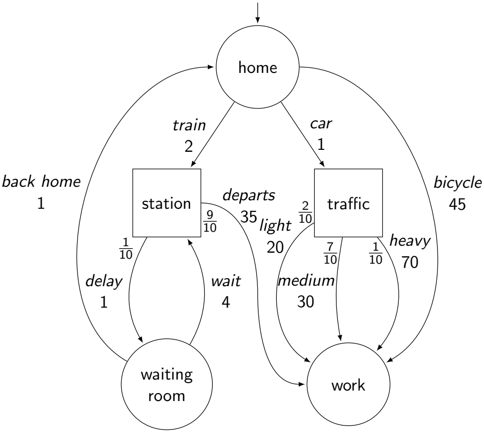


Example: going to work



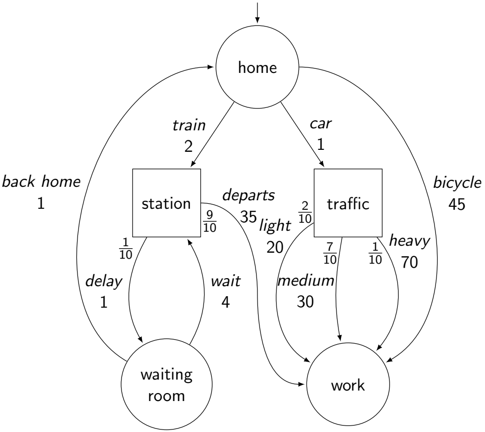
- Weights = minutes
- Goal: *minimize our expected time* to reach "work"
- **But**, important meeting in one hour! Requires *strict guarantees* on the worst-case reaching time.

Example: going to work



- ▷ Optimal expectation strategy: car.
  - $\mathbb{E} = 33$, $\mathrm{WC} = 71 > 60$.
- ▷ Optimal worst-case strategy: bicycle.
  - $\mathbb{E} = \mathrm{WC} = 45 < 60$.

Example: going to work



- ▷ Optimal expectation strategy: car.
  - $\mathbb{E} = 33$, WC $= 71 > 60$.
- ▷ Optimal worst-case strategy: bicycle.
  - $\mathbb{E} =$ WC $= 45 < 60$.
- ▷ **Sample BWC strategy**: try train up to 3 delays then switch to bicycle.
  - $\mathbb{E} \approx 37.56$, WC $= 59 < 60$.
  - Optimal $\mathbb{E}$ under WC constraint
  - Uses finite **memory**

Beyond worst-case synthesis

Formal definition

Given

- $G = (\mathcal{G}, S_1, S_2)$, $s_{\mathsf{init}} \in S$,
- a finite-memory stochastic model $\lambda_2^{\mathsf{stoch}} \in \Lambda_2^F$ of the adversary,
- a measurable value function $f\colon \mathsf{Plays}(\mathcal{G}) \rightarrow \mathbb{R} \cup \{-\infty, \infty\}$, and two thresholds $\mu, \nu \in \mathbb{Q}$,

the *beyond worst-case (BWC) problem* asks to decide if $\mathcal{P}_1$ has a finite-memory strategy $\lambda_1 \in \Lambda_1^F$ such that

$$\begin{cases} \forall\, \lambda_2 \in \Lambda_2, \forall\, \pi \in \mathsf{Outs}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2),\ f(\pi) > \mu & (1) \\ \mathbb{E}_{s_{\mathsf{init}}}^{G[\lambda_1, \lambda_2^{\mathsf{stoch}}]}(f) > \nu & (2) \end{cases}$$

and the *BWC synthesis problem* asks to synthesize such a strategy if one exists.

Beyond worst-case synthesis

Formal definition

Given

- $G = (\mathcal{G}, S_1, S_2)$, $s_{\mathsf{init}} \in S$,
- a finite-memory stochastic model $\lambda_2^{\mathsf{stoch}} \in \Lambda_2^F$ of the adversary,
- a measurable value function $f\colon \mathsf{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$, and two thresholds $\mu, \nu \in \mathbb{Q}$,

the *beyond worst-case (BWC) problem* asks to decide if $\mathcal{P}_1$ has a finite-memory strategy $\lambda_1 \in \Lambda_1^F$ such that

$$\begin{cases} \forall\, \lambda_2 \in \Lambda_2, \forall\, \pi \in \mathsf{Outs}_G(s_{\mathsf{init}}, \lambda_1, \lambda_2),\ f(\pi) > \mu & (1) \\ \mathbb{E}_{s_{\mathsf{init}}}^{G[\lambda_1, \lambda_2^{\mathsf{stoch}}]}(f) > \nu & (2) \end{cases}$$

and the *BWC synthesis problem* asks to synthesize such a strategy if one exists.

Notice the highlighted parts

Related work

Common philosophy: avoiding outlier outcomes

1. Our strategies are *strongly risk averse*
   - avoid risk at all costs and optimize among safe strategies

Related work

Common philosophy: avoiding outlier outcomes

1. Our strategies are *strongly risk averse*
   - ▷ avoid risk at all costs and optimize among safe strategies
2. Other notions of risk ensure low probability of risked behavior [WL99, FKR95]
   - ▷ without worst-case guarantee
   - ▷ without good expectation

Related work

Common philosophy: avoiding outlier outcomes

1. Our strategies are *strongly risk averse*
   - ▷ avoid risk at all costs and optimize among safe strategies
2. Other notions of risk ensure low probability of risked behavior [WL99, FKR95]
   - ▷ without worst-case guarantee
   - ▷ without good expectation
3. Trade-off between expectation and variance [BCFK13, MT11]
   - ▷ statistical measure of the stability of the performance
   - ▷ no strict guarantee on individual outcomes

Mean-payoff value function

	worst-case	expected value	**BWC**
complexity	$NP \cap coNP$	P	**$NP \cap coNP$**
memory	memoryless	memoryless	**pseudo-polynomial**

▷ Additional modeling power **for free**

Philosophy of the algorithm

- Classical worst-case and expected value results and algorithms as *nuts and bolts*
- *Screw them together* in an adequate way

Philosophy of the algorithm

- Classical worst-case and expected value results and algorithms as *nuts and bolts*
- *Screw them together* in an adequate way

Three key ideas

1. To characterize the expected value, look at *end-components* (ECs)
2. *Winning ECs* vs. *losing ECs*: the latter must be avoided to preserve the worst-case requirement
3. *Inside a WEC*, we have an interesting way to play...

Philosophy of the algorithm

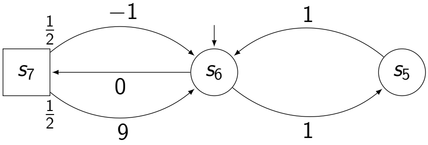
- Classical worst-case and expected value results and algorithms as *nuts and bolts*
- *Screw them together* in an adequate way

Three key ideas

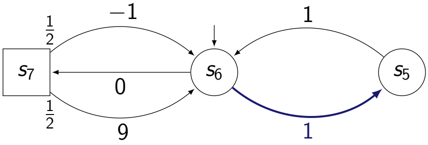
1. To characterize the expected value, look at *end-components* (ECs)
2. *Winning ECs* vs. *losing ECs*: the latter must be avoided to preserve the worst-case requirement
3. *Inside a WEC*, we have an interesting way to play...

$\Longrightarrow$ **Let's focus on a WEC**

Inside a WEC



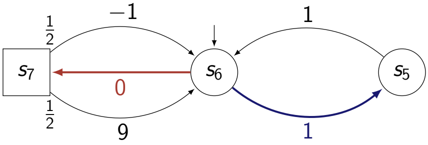
Inside a WEC



Game interpretation

- Worst-case threshold is $\mu = 0$
- **All** states are winning: memoryless optimal worst-case strategy $\lambda_1^{wc} \in \Lambda_1^{PM}(G)$, ensuring $\mu^* = 1 > 0$

Inside a WEC



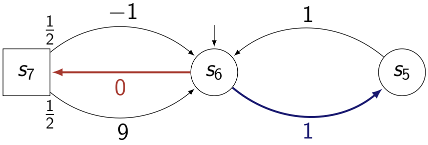
Game interpretation

- Worst-case threshold is $\mu = 0$
- **All** states are winning: memoryless optimal worst-case strategy $\lambda_1^{wc} \in \Lambda_1^{PM}(G)$, ensuring $\mu^* = 1 > 0$

MDP interpretation

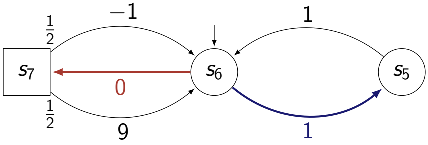
- Memoryless optimal expected value strategy $\lambda_1^{e} \in \Lambda_1^{PM}(P)$ achieves $\nu^* = 2$

A cornerstone of our approach



BWC problem: **what kind of thresholds $(\mu = 0, \nu)$ can we achieve?**

A cornerstone of our approach



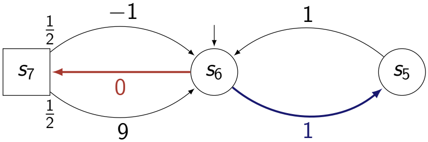
BWC problem: **what kind of thresholds $(\mu = 0, \nu)$ can we achieve?**

Key result

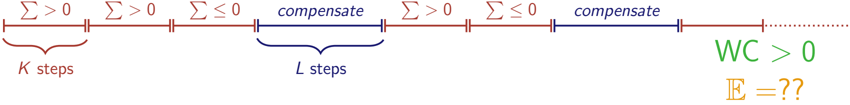
For all $\varepsilon > 0$, there exists a finite-memory strategy of $\mathcal{P}_1$ that satisfies the BWC problem for the thresholds pair $(0, \nu^* - \varepsilon)$.

- We can be **arbitrarily close to the optimal expectation** while ensuring the worst-case

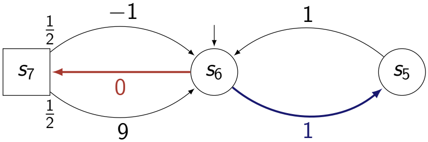
Combined strategy



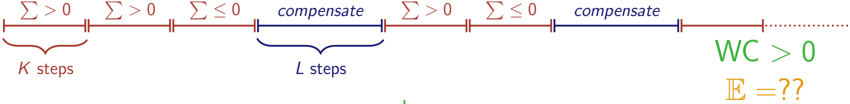
Outcomes of the form



Combined strategy



Outcomes of the form

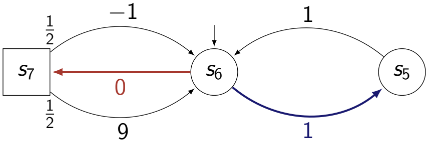


What we want

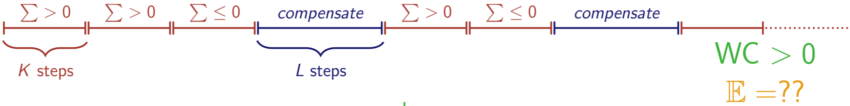
$K, L \to \infty$

$\mathbb{E} = \nu^* = 2$

Combined strategy



Outcomes of the form



What we want

$K, L \to \infty$

$L = \text{linear}(K)$

$\mathbb{P}(\longmapsto) \to 0$ exp. fast! [Tra09, GO02]

Shortest path

- Strictly positive integer weights, $w\colon E \to \mathbb{N}_0$
- $\mathcal{P}_1$ wants to minimize its total cost up to target
  - ▷ inequalities are reversed

Shortest path

- Strictly positive integer weights, $w\colon E \to \mathbb{N}_0$
- $\mathcal{P}_1$ wants to minimize its total cost up to target
  - ▷ inequalities are reversed

	worst-case	expected value	**BWC**
complexity	P	P	**pseudo-poly. / NP-hard**
memory	memoryless	memoryless	**pseudo-poly.**

▷ Problem **inherently harder** than worst-case and expectation.

▷ NP-hardness by K^{th} largest subset problem [JK78, GJ79]

Key difference with MP case

Useful observation

The set of all worst-case winning strategies for the shortest path can be represented through a finite game.

Sequential approach solving the BWC problem:

1. represent all WC winning strategies,
2. optimize the expected value within those strategies.

Multi-dimension games

<table>
<tr><th></th><th></th><th>EG</th><th>$\underline{\text{MP}}$</th><th>$\overline{\text{MP}}$</th><th>$\underline{\text{TP}}$</th><th>$\overline{\text{TP}}$</th></tr>
<tr><td rowspan="3">one-dim.</td><td>complexity</td><td colspan="5">NP ∩ coNP</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td colspan="5" rowspan="2">pure memoryless</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td></tr>
<tr><td rowspan="3">k-dim.</td><td>complexity</td><td colspan="2">coNP-c.</td><td>NP ∩ coNP</td><td colspan="2">??</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td>pure finite</td><td colspan="2">pure infinite</td><td colspan="2" rowspan="2">??</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td><td colspan="3">pure memoryless</td></tr>
</table>

▷ Natural extension

Multi-dimension games

<table>
<tr><th></th><th></th><th>EG</th><th>$\underline{\mathsf{MP}}$</th><th>$\overline{\mathsf{MP}}$</th><th>$\underline{\mathsf{TP}}$</th><th>$\overline{\mathsf{TP}}$</th></tr>
<tr><td rowspan="3">one-dim.</td><td>complexity</td><td colspan="5">NP $\cap$ coNP</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td colspan="5" rowspan="2">pure memoryless</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td></tr>
<tr><td rowspan="3">k-dim.</td><td>complexity</td><td colspan="2">coNP-c.</td><td>NP $\cap$ coNP</td><td colspan="2">??</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td>pure finite</td><td colspan="2">pure infinite</td><td colspan="2" rowspan="2">??</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td><td colspan="3">pure memoryless</td></tr>
</table>

▷ Natural extension, increased complexity.

Multi-dimension games

<table>
<tr><th></th><th></th><th>EG</th><th>$\underline{\text{MP}}$</th><th>$\overline{\text{MP}}$</th><th>$\underline{\text{TP}}$</th><th>$\overline{\text{TP}}$</th></tr>
<tr><td rowspan="3">one-dim.</td><td>complexity</td><td colspan="5">NP $\cap$ coNP</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td colspan="5" rowspan="2">pure memoryless</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td></tr>
<tr><td rowspan="3">k-dim.</td><td>complexity</td><td colspan="2">coNP-c.</td><td>NP $\cap$ coNP</td><td colspan="2">??</td></tr>
<tr><td>$\mathcal{P}_1$ mem.</td><td>pure finite</td><td colspan="2">pure infinite</td><td colspan="2" rowspan="2">??</td></tr>
<tr><td>$\mathcal{P}_2$ mem.</td><td colspan="3">pure memoryless</td></tr>
</table>

- ▷ Natural extension, increased complexity.
- ▷ **Question**: what about TP?

Multi-dimension games

		EG	$\underline{\text{MP}}$	$\overline{\text{MP}}$	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	NP $\cap$ coNP				
	$\mathcal{P}_1$ mem.	pure memoryless				
	$\mathcal{P}_2$ mem.					
k-dim.	complexity	coNP-c.		NP $\cap$ coNP	**undec.**	
	$\mathcal{P}_1$ mem.	pure finite	pure infinite		-	
	$\mathcal{P}_2$ mem.	pure memoryless				

Theorem

Total-payoff games are **undecidable** for $k \geq 5$.

- ▷ Reduction from the halting problem in 2CMs.
- ▷ Open for $k = 2, 3$ and 4.

Multi-dimension games

		EG	$\underline{\text{MP}}$	$\overline{\text{MP}}$	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	$\text{NP} \cap \text{coNP}$				
	$\mathcal{P}_1$ mem.	pure memoryless				
	$\mathcal{P}_2$ mem.					
k-dim.	complexity	coNP-c.		$\text{NP} \cap \text{coNP}$	undec.	
	$\mathcal{P}_1$ mem.	pure finite	pure infinite		-	
	$\mathcal{P}_2$ mem.	pure memoryless				

▷ We want **finite-memory** controllers.

Multi-dimension games

		EG	MP	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	$\text{NP} \cap \text{coNP}$			
	$\mathcal{P}_1$ mem.	pure memoryless			
	$\mathcal{P}_2$ mem.				
k-dim.	complexity	coNP-c.		undec.	
	$\mathcal{P}_1$ mem.	pure finite		-	
	$\mathcal{P}_2$ mem.	pure memoryless			

- ▷ We want **finite-memory** controllers.
- ▷ Restrict $\mathcal{P}_1$ to finite-memory strategies.

Lemma [CDHR10, VCD+12]

The answer to the worst-case mean-payoff threshold problem is YES iff the answer to the unknown initial credit problem is YES.

Multi-dimension games

		EG	MP	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	NP ∩ coNP			
	$\mathcal{P}_1$ mem.	pure memoryless			
	$\mathcal{P}_2$ mem.				
k-dim.	complexity	coNP-c.		undec.	
	$\mathcal{P}_1$ mem.	pure finite		-	
	$\mathcal{P}_2$ mem.	pure memoryless			

- **Question**: precise **memory bounds**?
  - ▷ **exponential** memory sufficient and necessary

Multi-dimension games

		EG	MP	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	NP ∩ coNP			
	$\mathcal{P}_1$ mem.	pure memoryless			
	$\mathcal{P}_2$ mem.				
k-dim.	complexity	coNP-c.		undec.	
	$\mathcal{P}_1$ mem.	pure finite		-	
	$\mathcal{P}_2$ mem.	pure memoryless			

- **Question**: precise **memory bounds**?
  - ▷ **exponential** memory sufficient and necessary
- **Question**: efficient **synthesis algorithm**?
  - ▷ **EXPTIME** algorithm
  - ▷ symbolic and incremental

Multi-dimension games

		EG	MP	$\underline{\text{TP}}$	$\overline{\text{TP}}$
one-dim.	complexity	$\text{NP} \cap \text{coNP}$			
	$\mathcal{P}_1$ mem.	pure memoryless			
	$\mathcal{P}_2$ mem.				
k-dim.	complexity	coNP-c.		undec.	
	$\mathcal{P}_1$ mem.	pure finite		-	
	$\mathcal{P}_2$ mem.	pure memoryless			

- **Question**: precise **memory bounds**?
  - ▷ **exponential** memory sufficient and necessary
- **Question**: efficient **synthesis algorithm**?
  - ▷ **EXPTIME** algorithm
  - ▷ symbolic and incremental
- Results for EG / MP + **parity**.

Trading finite memory for randomness

Question: when and how can $\mathcal{P}_1$ trade his pure finite-memory strategy for an equally powerful randomized memoryless one?

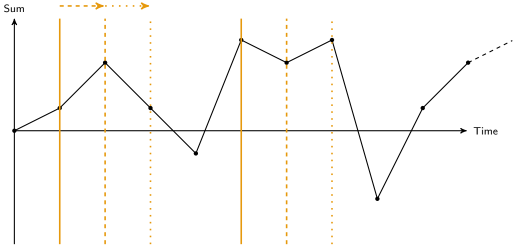
- ▷ relax to *almost-sure* semantics

	Multi energy and energy parity	Multi MP (parity)	MP parity
one-player	×	√	√
two-player	×	×	√

Why an alternative to MP/TP?

- No known polynomial-time algorithm in one-dimension.
- TP is undecidable in multi-dimension.
- No timing guarantee
  - ▷ long-run behavior vs. time frames.

Window objectives: key idea



- **Window** of fixed size **sliding** along a play
  $\rightsquigarrow$ defines a local finite horizon.
- Objective: see a **local** $MP \geq 0$ *before hitting the end* of the window
  $\rightsquigarrow$ needs to be verified at *every* step.

▷ Intuition: local deviations from the threshold must be compensated in a parametrized # of steps.

Multiple variants

- Maximal window size fixed or quantified existentially (**B**ounded **W**indow)
- Prefix-independent or not

Multiple variants

- Maximal window size fixed or quantified existentially (**B**ounded **W**indow)
- Prefix-independent or not

Conservative approximations in one-dim.

$$\text{Any window obj.} \Rightarrow \textbf{BW} \Rightarrow \text{MP} \geq 0$$
$$\textbf{BW} \Leftarrow \text{MP} > 0$$

Results overview

	one-dimension			k-dimension		
	complexity	$\mathcal{P}_1$ mem.	$\mathcal{P}_2$ mem.	complexity	$\mathcal{P}_1$ mem.	$\mathcal{P}_2$ mem.
$\underline{\text{MP}}$ / $\overline{\text{MP}}$	NP $\cap$ coNP	mem-less		coNP-c. / NP $\cap$ coNP	infinite	mem-less
$\underline{\text{TP}}$ / $\overline{\text{TP}}$	NP $\cap$ coNP	mem-less		**undec.**	-	-
WMP: fixed polynomial window	**P-c.**	**mem. req.** $\leq$ **linear**$(\lvert S\rvert \cdot l_{\text{max}})$		**PSPACE-h.** **EXP-easy**	**exponential**	
WMP: fixed arbitrary window	**P**$(\lvert S\rvert, V, l_{\text{max}})$			**EXP-c.**		
WMP: bounded window problem	**NP $\cap$ coNP**	**mem-less**	**infinite**	**NPR-h.**	-	-

- $\lvert S\rvert$ the # of states, V the length of the binary encoding of weights, and l_{max} the window size.

Results overview: advantages

	one-dimension			k-dimension		
	complexity	$\mathcal{P}_1$ mem.	$\mathcal{P}_2$ mem.	complexity	$\mathcal{P}_1$ mem.	$\mathcal{P}_2$ mem.
$\underline{\text{MP}}$ / $\overline{\text{MP}}$	$\text{NP} \cap \text{coNP}$	mem-less		coNP-c. / $\text{NP} \cap \text{coNP}$	infinite	mem-less
$\underline{\text{TP}}$ / $\overline{\text{TP}}$	$\text{NP} \cap \text{coNP}$	mem-less		**undec.**	-	-
WMP: fixed polynomial window	**P-c.**	**mem. req.** $\leq$ **linear**$(\lvert S\rvert \cdot l_{max})$		**PSPACE-h.** **EXPTIME-easy**	**exponential**	
WMP: fixed arbitrary window	**P**$(\lvert S\rvert, V, l_{max})$			**EXPTIME-c.**		
WMP: bounded window problem	$\mathbf{NP} \cap \mathbf{coNP}$	**mem-less**	**infinite**	**NPR-h.**	-	-

- $\lvert S\rvert$ the # of states, V the length of the binary encoding of weights, and l_{max} the window size.
- For one-dim. games with poly. windows, we are in **P**.
- For multi-dim. games with fixed windows, we are **decidable**.
- Window obj. provide **timing guarantees**.

Summary

Study of *multi-criteria* quantitative games.

Summary

Study of *multi-criteria* quantitative games.

1. **Beyond worst-case synthesis**
   - worst-case *and* expected value
   - additional modeling power for free in MP case
   - complexity leap for SP

Summary

Study of *multi-criteria* quantitative games.

1. **Beyond worst-case synthesis**
   - worst-case *and* expected value
   - additional modeling power for free in MP case
   - complexity leap for SP
2. **Multi-dimension TP, MP and EG + parity**
   - undecidability of TP
   - tight memory bounds for MP and EG + parity
   - optimal synthesis algorithm
   - memory vs. randomness

Summary

Study of *multi-criteria* quantitative games.

1. **Beyond worst-case synthesis**
   - worst-case *and* expected value
   - additional modeling power for free in MP case
   - complexity leap for SP
2. **Multi-dimension TP, MP and EG + parity**
   - undecidability of TP
   - tight memory bounds for MP and EG + parity
   - optimal synthesis algorithm
   - memory vs. randomness
3. **Window objectives**
   - timing guarantees
   - improved tractability

Future work

- **Beyond worst-case extensions**
  - ▷ more general games (e.g., stochastic games)
  - ▷ multi-dimension
  - ▷ percentile performances

Future work

- **Beyond worst-case extensions**
  - ▷ more general games (e.g., stochastic games)
  - ▷ multi-dimension
  - ▷ percentile performances
- **Mixed objectives**

Future work

- **Beyond worst-case extensions**
  - ▷ more general games (e.g., stochastic games)
  - ▷ multi-dimension
  - ▷ percentile performances
- **Mixed objectives**
- **Window objectives**
  - ▷ stochastic context
  - ▷ synchronous closing
  - ▷ (finitary) parity [CHH09]

Thanks!

To my advisors, Véronique Bruyère and Jean-François Raskin,

to my other co-authors, Krishnendu Chatterjee, Laurent Doyen and Emmanuel Filiot,

and to you, members of the jury, for your precious time!

References I

Tomás Brázdil, Krishnendu Chatterjee, Vojtech Forejt, and Antonín Kucera.
Trading performance for stability in Markov decision processes.
In LICS, pages 331–340. IEEE Computer Society, 2013.

Véronique Bruyère, Emmanuel Filiot, Mickael Randour, and Jean-François Raskin.
Expectations or guarantees? i want it all! a crossroad between games and mdps.
In Fabio Mogavero, Aniello Murano, and Moshe Y. Vardi, editors, Proceedings 2nd International Workshop on Strategic Reasoning, Grenoble, France, April 5-6, 2014, volume 146 of Electronic Proceedings in Theoretical Computer Science, pages 1–8. Open Publishing Association, 2014.

Véronique Bruyère, Emmanuel Filiot, Mickael Randour, and Jean-François Raskin.
Meet Your Expectations With Guarantees: Beyond Worst-Case Synthesis in Quantitative Games.
In Ernst W. Mayr and Natacha Portier, editors, 31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), volume 25 of Leibniz International Proceedings in Informatics (LIPIcs), pages 199–213, Dagstuhl, Germany, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, and Jean-François Raskin.
Generalized mean-payoff and energy games.
In Kamal Lodaya and Meena Mahajan, editors, FSTTCS, 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.
In Dang Van Hung and Mizuhito Ogawa, editors, ATVA, volume 8172 of Lecture Notes in Computer Science, pages 118–132. Springer, 2013.

References II

Krishnendu Chatterjee, Thomas A. Henzinger, and Florian Horn.
Finitary winning in omega-regular games.
ACM Trans. Comput. Log., 11(1), 2009.

Krishnendu Chatterjee, Mickael Randour, and Jean-François Raskin.
Strategy synthesis for multi-dimensional quantitative objectives.
In Maciej Koutny and Irek Ulidowski, editors, CONCUR, volume 7454 of Lecture Notes in Computer Science, pages 115–131. Springer, 2012.

Krishnendu Chatterjee, Mickael Randour, and Jean-François Raskin.
Strategy synthesis for multi-dimensional quantitative objectives.
Acta Informatica, 2013.
35 pages.

Jerzy A. Filar, Dmitry Krass, and Kirsten W. Ross.
Percentile performance criteria for limiting average Markov decision processes.
Transactions on Automatic Control, 40(1):2–10, 1995.

Michael R. Garey and David S. Johnson.
Computers and intractability: a guide to the Theory of NP-Completeness.
Freeman New York, 1979.

Peter W. Glynn and Dirk Ormoneit.
Hoeffding's inequality for uniformly ergodic Markov chains.
Statistics & Probability Letters, 56(2):143–146, 2002.

References III

Donald B. Johnson and Samuel D. Kashdan.
Lower bounds for selection in X + Y and other multisets.
Journal of the ACM, 25(4):556–570, 1978.

Shie Mannor and John N. Tsitsiklis.
Mean-variance optimization in Markov decision processes.
In Lise Getoor and Tobias Scheffer, editors, ICML, pages 177–184. Omnipress, 2011.

Mickael Randour.
Automated synthesis of reliable and efficient systems through game theory: A case study.
In Thomas Gilbert, Markus Kirkilionis, and Gregoire Nicolis, editors, Proceedings of the European Conference on Complex Systems 2012, Springer Proceedings in Complexity, pages 731–738. Springer, 2013.

Mathieu Tracol.
Fast convergence to state-action frequency polytopes for MDPs.
Oper. Res. Lett., 37(2):123–126, 2009.

Yaron Velner, Krishnendu Chatterjee, Laurent Doyen, Thomas A. Henzinger, Alexander Rabinovich, and Jean-François Raskin.
The complexity of multi-mean-payoff and multi-energy games.
CoRR, abs/1209.3234, 2012.

Congbin Wu and Yuanlie Lin.
Minimizing risk models in Markov decision processes with policies depending on target values.
Journal of Mathematical Analysis and Applications, 231(1):47–67, 1999.