Synthesis in Multi-Criteria Quantitative Games

Mickael Randour

LSV - CNRS & ENS Cachan, France

September 9, 2015 - *CSL 2015, Berlin*

Ackermann Award Lecture

Thesis: synthesis in multi-criteria quantitative games

- ▷ University of Mons, Belgium, April 2014.
- ▷ Currently post-doc in LSV, France, with Patricia Bouyer and Nicolas Markey.

Véronique Bruyère
UMONS, Belgium

Advisors

Jean-François Raskin
ULB, Belgium

Thesis: synthesis in multi-criteria quantitative games

- University of Mons, Belgium, April 2014.
- Currently post-doc in LSV, France, with Patricia Bouyer and Nicolas Markey.

Véronique Bruyère
UMONS, Belgium

Advisors

Jean-François Raskin
ULB, Belgium

Aim of this talk

Sketch the **motivation** for the research and give some **examples** of the studied problems.

General context

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

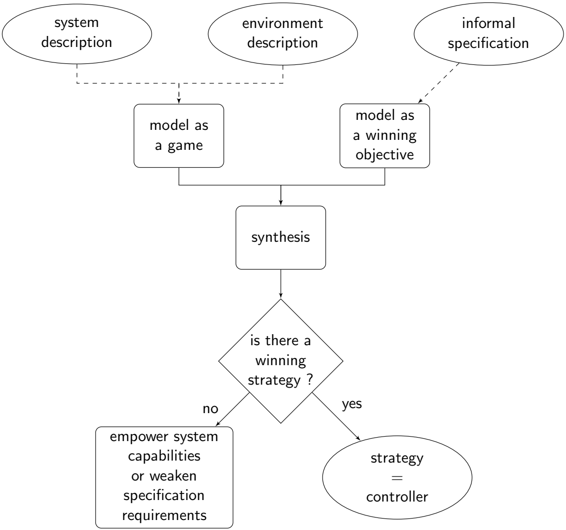
- Automated **controller synthesis** via games.

General context

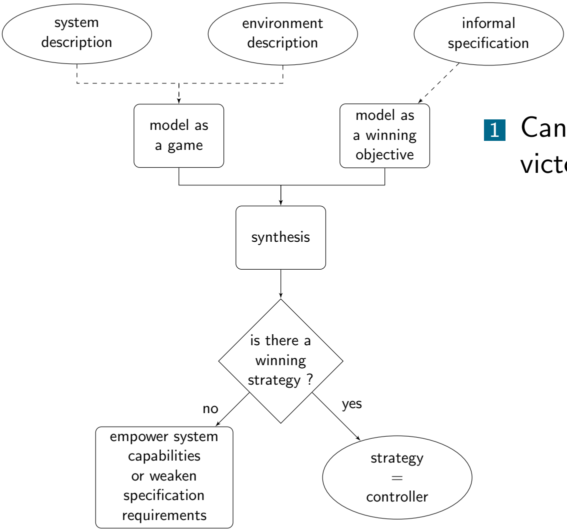
- Verification and synthesis:
  - ▷ a reactive **system** to *control*,
  - ▷ an *interacting* **environment**,
  - ▷ a **specification** to *enforce*.
- Automated **controller synthesis** via games.
- Strong links between logic and games:
  - ▷ logic used as specification language,
  - ▷ model checking via game solving (e.g., parity games for modal μ-calculus).

Here: focus on the game-theoretic view.

Synthesis via two-player graph games

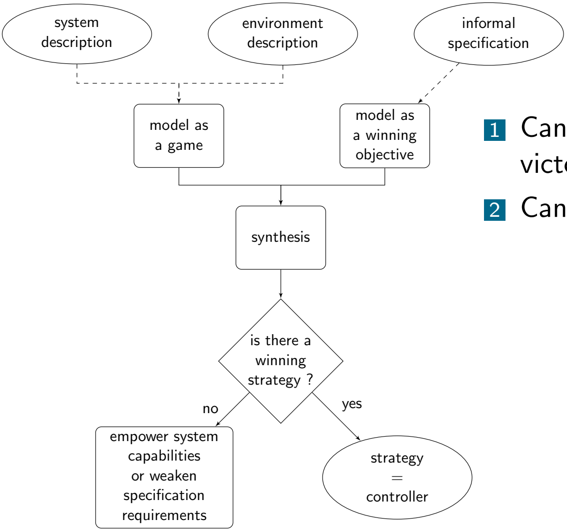


Synthesis via two-player graph games



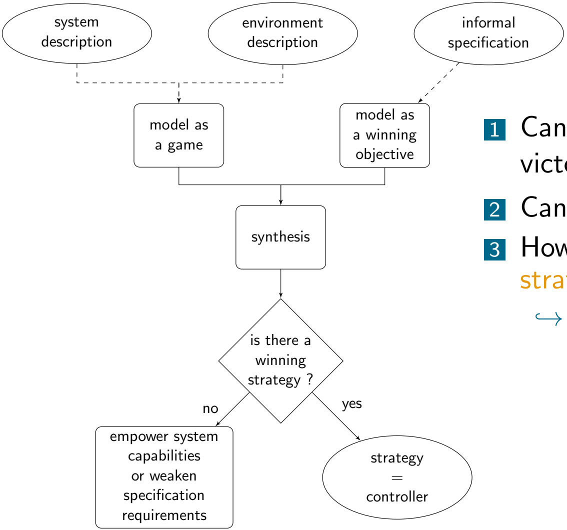
1. Can one player guarantee victory?

Synthesis via two-player graph games



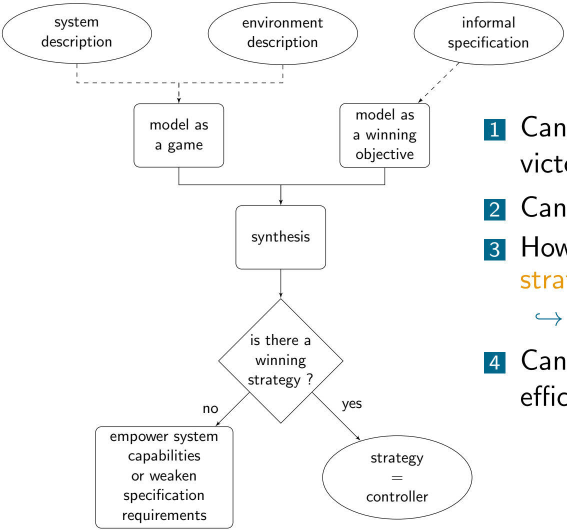
1. Can one player guarantee victory?
2. Can we decide which one?

Synthesis via two-player graph games



1. Can one player guarantee victory?
2. Can we decide which one?
3. How complex his winning strategy needs to be?
   $\hookrightarrow$ **Simpler is better.**

Synthesis via two-player graph games



1. Can one player guarantee victory?
2. Can we decide which one?
3. How complex his winning strategy needs to be?
   ↪ **Simpler is better.**
4. Can we synthesize one efficiently?

From Boolean to quantitative and beyond: an ongoing shift

Boolean view: behavior is either correct or incorrect.
No interpretation of *how good* it is.
OK for yes-no properties (e.g., no deadlock).
Example: parity games [GTW02, EJS93, Jur98].

From Boolean to quantitative and beyond: an ongoing shift

Boolean view: behavior is either correct or incorrect.
No interpretation of *how good* it is.
OK for yes-no properties (e.g., no deadlock).
Example: parity games [GTW02, EJS93, Jur98].

↓

Quantitative view: rank the performance, model resource constraints.
Traditionally, only *single-criterion* models.
OK for energy consumption, response time [CdAHS03, BCHJ09, Ran13].
Example: mean-payoff games [EM79, ZP96, BCD$^+$11].

From Boolean to quantitative and beyond: an ongoing shift

Boolean view: behavior is either correct or incorrect.
No interpretation of *how good* it is.
OK for yes-no properties (e.g., no deadlock).
Example: parity games [GTW02, EJS93, Jur98].

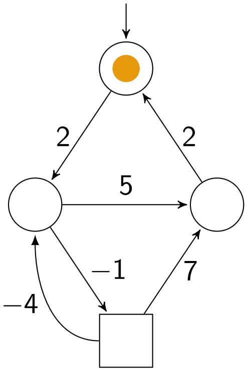
↓

Quantitative view: rank the performance, model resource constraints.
Traditionally, only *single-criterion* models.
OK for energy consumption, response time [CdAHS03, BCHJ09, Ran13].
Example: mean-payoff games [EM79, ZP96, BCD+11].

↓

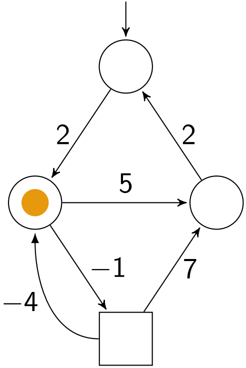
Multi-criteria view: study interplays and trade-offs.
E.g., response time vs. computing power vs. energy consumption.
Also, consider strategies with richer guarantees.
E.g., average performance vs. worst-case performance.

Quantitative games on graphs



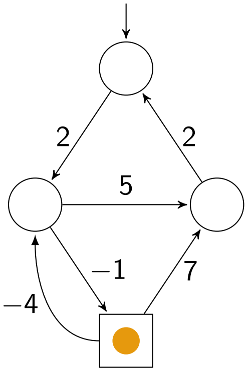
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- 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



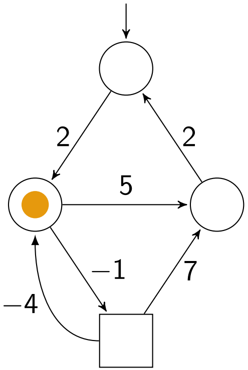
- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- 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



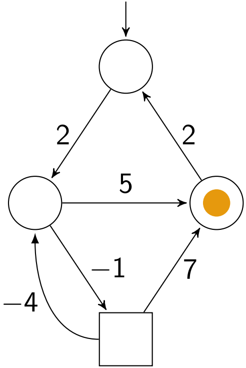
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- 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



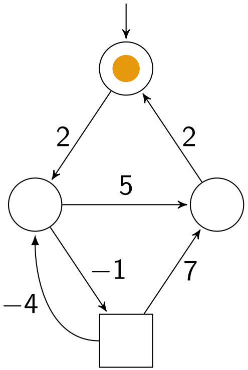
- Graph $\mathcal{G} = (S, E, w)$ with $w \colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f \colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- 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



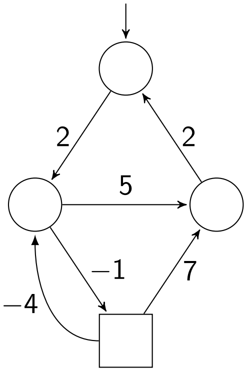
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
- 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 $\mathcal{M}(\lambda_i) = (\text{Mem}, \text{m}_0, \alpha_{\text{u}}, \alpha_{\text{n}})$

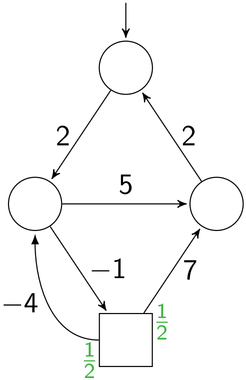
Quantitative games on graphs



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

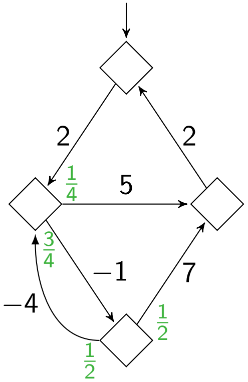
- Graph $\mathcal{G} = (S, E, w)$ with $w\colon E \to \mathbb{Z}$
- Deterministic transitions
- Two-player *game* $G = (\mathcal{G}, S_1, S_2)$
  - ▷ $\mathcal{P}_1$ states = ○
  - ▷ $\mathcal{P}_2$ states = □
- Plays have values
  - ▷ $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty,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

Markov decision processes



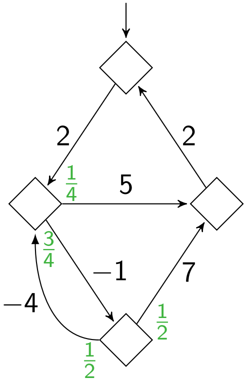
- MDP $P = (\mathcal{G}, S_1, S_\Delta, \Delta)$ with $\Delta\colon S_\Delta \to \mathcal{D}(S)$
  - ▷ $\mathcal{P}_1$ states = ◯
  - ▷ stochastic states = □
- MDP = game + strategy of $\mathcal{P}_2$
  - ▷ $P = G[\lambda_2]$

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 $\mathcal{A} \subseteq \text{Plays}(\mathcal{G})$
  - ▷ probability $\mathbb{P}^M_{s_{\text{init}}}(\mathcal{A})$
- Measurable $f\colon \text{Plays}(\mathcal{G}) \to \mathbb{R} \cup \{-\infty, \infty\}$
  - ▷ *expected value* $\mathbb{E}^M_{s_{\text{init}}}(f)$

Winning semantics and decision problems

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

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

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

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

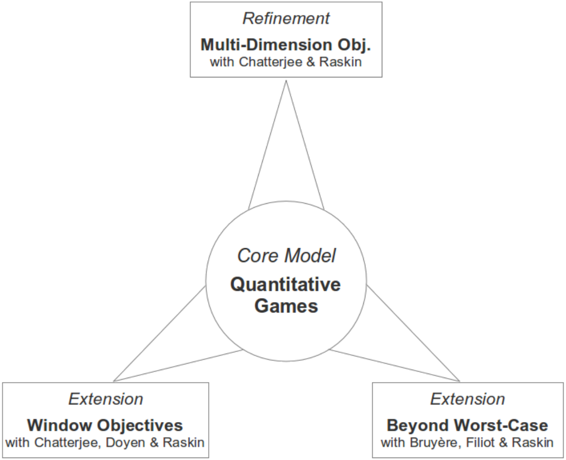
Single-criterion models - known results

		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			

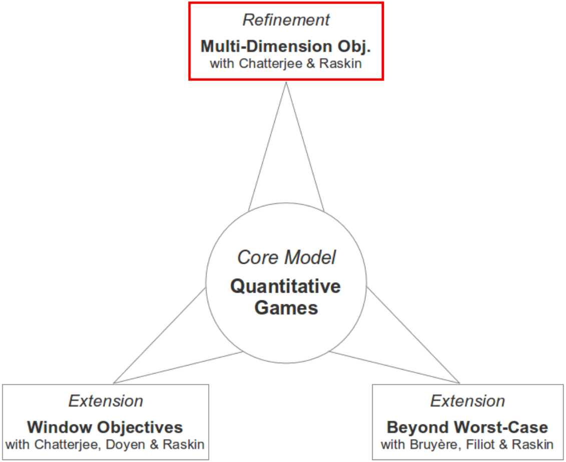
- ▷ Simple strategies suffice (no memory, no randomness).
- ▷ "Low" complexity but important open problem: $UP \cap coUP \rightsquigarrow P$?

Contributions



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

Multi-dimension objectives



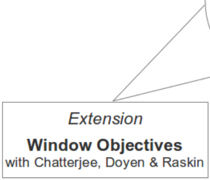
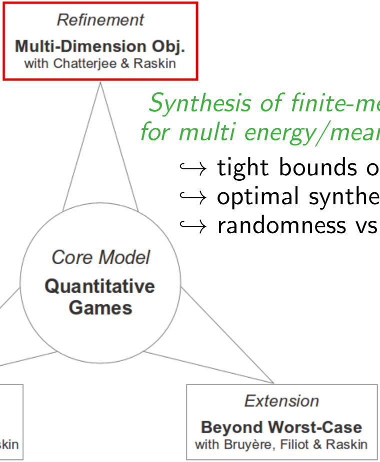
Weights = vectors $\rightsquigarrow$ multiple objectives $\rightsquigarrow$ **trade-offs**

Complexity $\nearrow$ (e.g., coNP-c. for solving energy games)

Multi-dimension objectives

K. Chatterjee
IST Austria

J.-F. Raskin
ULB

Synthesis of finite-memory strategies for multi energy/mean-payoff + parity

$\hookrightarrow$ tight bounds on memory
$\hookrightarrow$ optimal synthesis algorithm
$\hookrightarrow$ randomness vs. memory

Weights = vectors $\rightsquigarrow$ multiple objectives $\rightsquigarrow$ **trade-offs**

Complexity $\nearrow$ (e.g., coNP-c. for solving energy games)

Multi-dimension objectives

K. Chatterjee
IST Austria

J.-F. Raskin
ULB

L. Doyen
LSV, ENS Cachan

Refinement
Multi-Dimension Obj.
with Chatterjee & Raskin

Synthesis of finite-memory strategies for multi energy/mean-payoff + parity

- $\hookrightarrow$ tight bounds on memory
- $\hookrightarrow$ optimal synthesis algorithm
- $\hookrightarrow$ randomness vs. memory

Core Model
Quantitative Games

Undecidability of multi total-payoff

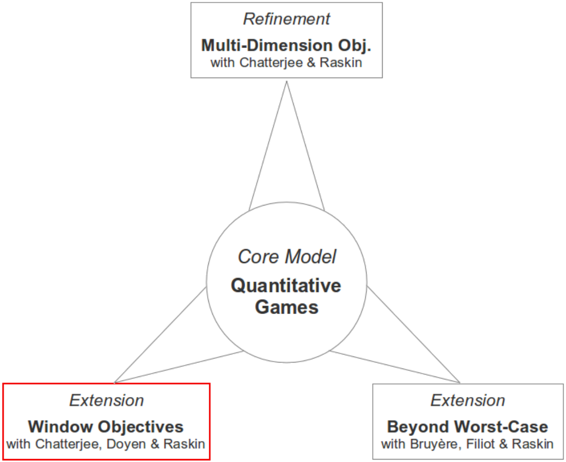
Extension
Window Objectives
with Chatterjee, Doyen & Raskin

Extension
Beyond Worst-Case
with Bruyère, Filiot & Raskin

Weights = vectors $\rightsquigarrow$ multiple objectives $\rightsquigarrow$ **trade-offs**

Complexity $\nearrow$ (e.g., coNP-c. for solving energy games)

Window objectives



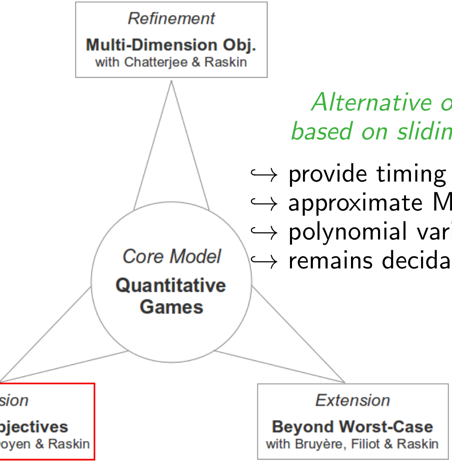
Mean-payoff and total-payoff have: **limited tractability** ($\in$ P ?? + multi TP undec.) and **no timing guarantee** (limit behavior)

Window objectives

K. Chatterjee
IST Austria

J.-F. Raskin
ULB

L. Doyen
LSV, ENS Cachan

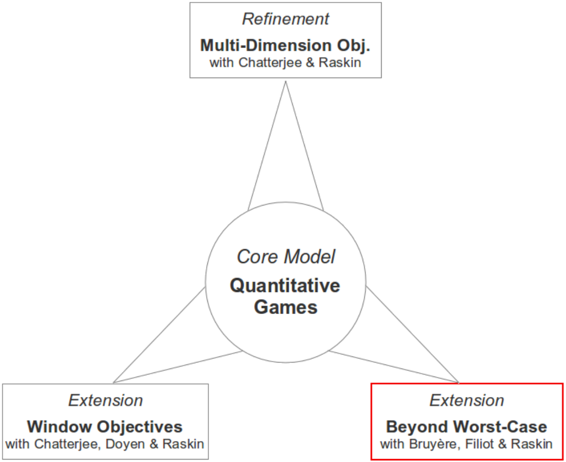


Alternative objectives based on sliding windows

- $\hookrightarrow$ provide timing guarantees
- $\hookrightarrow$ approximate MP and TP
- $\hookrightarrow$ polynomial variant in one-dim.
- $\hookrightarrow$ remains decidable in multi-dim.

Mean-payoff and total-payoff have: **limited tractability** ($\in$ P ?? + multi TP undec.) and **no timing guarantee** (limit behavior)

Beyond worst-case synthesis



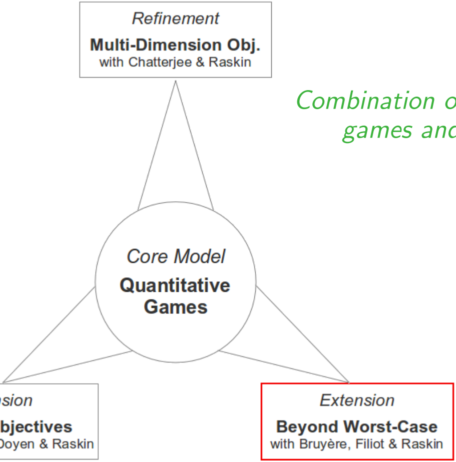
Framework for the analysis of performance trade-offs w.r.t. the **nature of the environment**.

Beyond worst-case synthesis

V. Bruyère
UMONS

E. Filiot
ULB

J.-F. Raskin
ULB



Combination of two views: games and MDPs

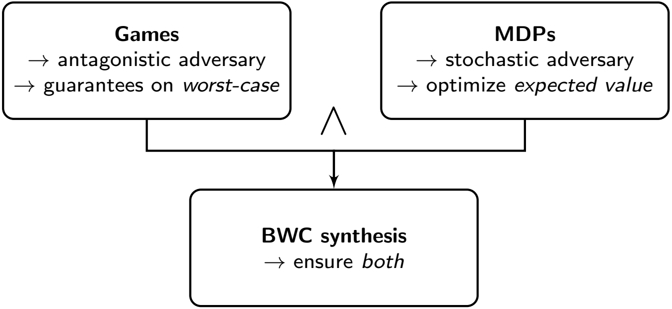
Framework for the analysis of performance trade-offs w.r.t. the **nature of the environment**.

Beyond worst-case synthesis

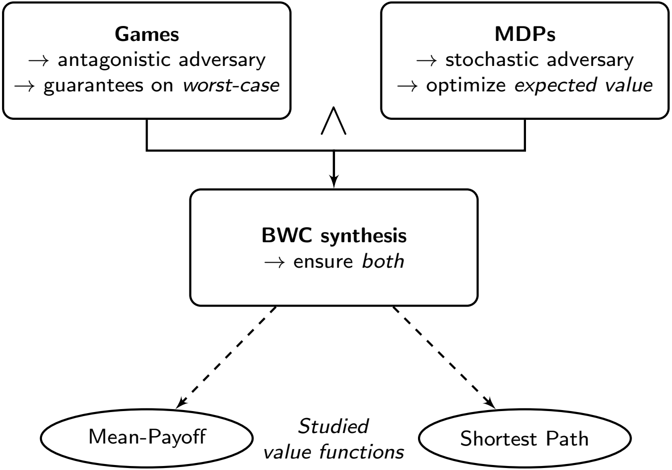
Games
→ antagonistic adversary
→ guarantees on *worst-case*

MDPs
→ stochastic adversary
→ optimize *expected value*

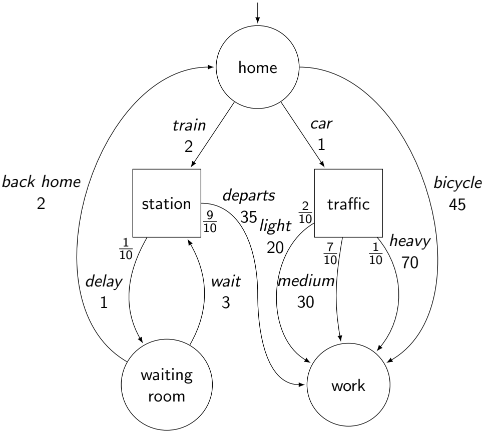
Beyond worst-case synthesis



Beyond worst-case synthesis

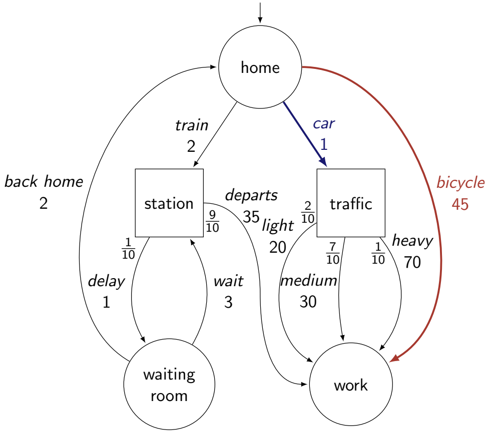


Example: going to work (shortest path)



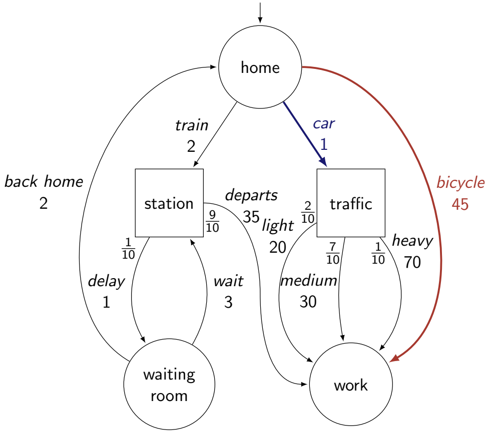
- Weights = minutes
- Goal: minimize our **expected time** to reach "work"
- **But**, important meeting in one hour! Requires **strict guarantees on the worst-case** reaching time.

Example: going to work (shortest path)



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

Example: going to work (shortest path)



- ▷ Optimal **expectation** strategy: take the car.
  - $\mathbb{E} = 33$, $\text{WC} = 71 > 60$.
- ▷ Optimal **worst-case** strategy: bicycle.
  - $\mathbb{E} = \text{WC} = 45 < 60$.
- ▷ **Sample BWC strategy**: try train up to 3 delays then switch to bicycle.
  - $\mathbb{E} \approx 37.45$, $\text{WC} = 58 < 60$.
  - Optimal $\mathbb{E}$ under WC constraint
  - Uses finite **memory**

BWC synthesis: overview

- **Mean-payoff**

	worst-case	expected value	**BWC**
complexity	$\mathsf{NP} \cap \mathsf{coNP}$	P-c.	**$\mathsf{NP} \cap \mathsf{coNP}$**
memory	pure memoryless		**pure pseudo-poly.**

▷ Additional modeling power **for free**!

▷ Constructing correct strategies require careful analysis and is technically involved.

BWC synthesis: overview

- **Mean-payoff**

	worst-case	expected value	**BWC**
complexity	$NP \cap coNP$	P-c.	$\mathbf{NP \cap coNP}$
memory	pure memoryless		**pure pseudo-poly.**

▷ Additional modeling power **for free**!

▷ Constructing correct strategies require careful analysis and is technically involved.

- **Shortest path**

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

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

Conclusion and future work

Key idea

We need innovative models to encompass the complexity of practical applications: trade-offs, strategies with rich guarantees...

Some challenges:

Conclusion and future work

Key idea

We need innovative models to encompass the complexity of practical applications: trade-offs, strategies with rich guarantees...

Some challenges:

- ▷ *Further extend our frameworks.*
  - ↪ Example: complex strategy profiling in multi-objective MDPs through percentile queries [RRS15a, RRS15b].

Conclusion and future work

Key idea

We need innovative models to encompass the complexity of practical applications: trade-offs, strategies with rich guarantees...

Some challenges:

- *Further extend our frameworks.*
  - ↪ Example: complex strategy profiling in multi-objective MDPs through percentile queries [RRS15a, RRS15b].
- *Full-fledged tool support.*
  - ↪ Some results led to integration in Acacia+ [BBFR13] and UPPAAL [DJL+14].

Conclusion and future work

Key idea

We need innovative models to encompass the complexity of practical applications: trade-offs, strategies with rich guarantees. . .

Some challenges:

- *Further extend our frameworks.*
  - ↪ Example: complex strategy profiling in multi-objective MDPs through percentile queries [RRS15a, RRS15b].
- *Full-fledged tool support.*
  - ↪ Some results led to integration in Acacia+ [BBFR13] and UPPAAL [DJL+14].
- *Mixed objectives.*

Conclusion and future work

Key idea

We need innovative models to encompass the complexity of practical applications: trade-offs, strategies with rich guarantees...

Some challenges:

- *Further extend our frameworks.*
  - ↪ Example: complex strategy profiling in multi-objective MDPs through percentile queries [RRS15a, RRS15b].
- *Full-fledged tool support.*
  - ↪ Some results led to integration in Acacia+ [BBFR13] and UPPAAL [DJL+14].
- *Mixed objectives.*
- *Work toward a unifying meta-framework.*
  - ↪ Seems difficult in full generality but still room to extract common underlying principles to instantiate in specific settings.

Thank you!

Any question?

References I

A. Bohy, V. Bruyère, E. Filiot, and J.-F. Raskin.
Synthesis from LTL specifications with mean-payoff objectives.
In Proc. of TACAS, LNCS 7795, pages 169–184. Springer, 2013.

L. Brim, J. Chaloupka, L. Doyen, R. Gentilini, and J.-F. Raskin.
Faster algorithms for mean-payoff games.
Formal Methods in System Design, 38(2):97–118, 2011.

R. Bloem, K. Chatterjee, T.A. Henzinger, and B. Jobstmann.
Better quality in synthesis through quantitative objectives.
In Proc. of CAV, LNCS 5643, pages 140–156. Springer, 2009.

V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin.
Expectations or guarantees? I want it all! A crossroad between games and MDPs.
In Proc. of SR, EPTCS 146, pages 1–8, 2014.

V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin.
Meet Your Expectations With Guarantees: Beyond Worst-Case Synthesis in Quantitative Games.
In Proc. of STACS, LIPIcs 25, pages 199–213. Sch. Dagstuhl–LZI, 2014.

V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin.
Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games.
Information and Computation, pages 1–44, 2015.
To appear.

References II

P. Bouyer, N. Markey, M. Randour, K.G. Larsen, and S. Laursen.
Average-energy games.
In Proc. of GandALF, EPTCS, 2015.

A. Chakrabarti, L. de Alfaro, T.A. Henzinger, and M. Stoelinga.
Resource interfaces.
In Proc. of EMSOFT, LNCS 2855, pages 117–133. Springer, 2003.

K. Chatterjee, L. Doyen, M. Randour, and J.-F. Raskin.
Looking at mean-payoff and total-payoff through windows.
In Proc. of ATVA, LNCS 8172, pages 118–132. Springer, 2013.

K. Chatterjee, L. Doyen, M. Randour, and J.-F. Raskin.
Looking at mean-payoff and total-payoff through windows.
Information and Computation, 242:25–52, 2015.

K. Chatterjee, M. Randour, and J.-F. Raskin.
Strategy synthesis for multi-dimensional quantitative objectives.
In Proc. of CONCUR, LNCS 7454, pages 115–131. Springer, 2012.

K. Chatterjee, M. Randour, and J.-F. Raskin.
Strategy synthesis for multi-dimensional quantitative objectives.
Acta Informatica, 51(3-4):129–163, 2014.

A. David, P.G. Jensen, K.G. Larsen, A. Legay, D. Lime, M.G. Sørensen, and J.H. Taankvist.
On time with minimal expected cost!
In Proc. of ATVA, LNCS 8837, pages 129–145. Springer, 2014.

References III

E.A. Emerson, C.S. Jutla, and A.P. Sistla.
On model-checking for fragments of μ-calculus.
In Proc. of CAV, LNCS 697, pages 385–396. Springer, 1993.

A. Ehrenfeucht and J. Mycielski.
Positional strategies for mean payoff games.
Int. Journal of Game Theory, 8(2):109–113, 1979.

E. Grädel, W. Thomas, and T. Wilke, editors.
Automata, Logics, and Infinite Games: A Guide to Current Research, LNCS 2500. Springer, 2002.

M. Jurdziński.
Deciding the winner in parity games is in UP $\cap$ co-UP.
Inf. Process. Lett., 68(3):119–124, 1998.

M. Randour.
Automated synthesis of reliable and efficient systems through game theory: A case study.
In Proc. of ECCS 2012, Springer Proceedings in Complexity XVII, pages 731–738. Springer, 2013.

M. Randour.
Synthesis in Multi-Criteria Quantitative Games.
PhD thesis, University of Mons, Belgium, 2014.

M. Randour, J.-F. Raskin, and O. Sankur.
Percentile queries in multi-dimensional markov decision processes.
In Proc. of CAV, LNCS 9206, pages 123–139. Springer, 2015.

References IV

M. Randour, J.-F. Raskin, and O. Sankur.
Variations on the stochastic shortest path problem.
In Proc. of VMCAI, LNCS 8931, pages 1–18. Springer, 2015.

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

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">$\mathsf{NP} \cap \mathsf{coNP}$</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>$\mathsf{NP} \cap \mathsf{coNP}$</td><td colspan="2">undec.</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>

Randomness instead of memory?

	Multi EG and EG parity	Multi MP (parity)	MP parity
one-player	$\times$	$\surd$	$\surd$
two-player	$\times$	$\times$	$\surd$

Window objectives

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