Planning a Journey in an Uncertain Environment: Variations on the Stochastic Shortest Path Problem

Mickael Randour
LSV - CNRS & ENS Cachan / Complexys, UMONS

13.05.2015

Annual meeting EDT Complex, UNamur
Controller synthesis

Setting:
- a reactive system to control,
- an interacting environment,
- a specification to enforce.
Controller synthesis

- Setting:
  - a reactive **system** to control,
  - an *interacting environment*,
  - a **specification** to enforce.

- For **critical** systems (e.g., airplane controller, power plants, ABS), testing is not enough!
  - ⇒ Need **formal methods**.
Controller synthesis

- Setting:
  - a reactive **system** to control,
  - an **interacting environment**,
  - a **specification** to enforce.

- For **critical** systems (e.g., airplane controller, power plants, ABS), testing is not enough!
  - ⇒ Need **formal methods**.

- **Automated synthesis** of provably-correct and efficient controllers:
  - mathematical frameworks,
    - e.g., game theory [GTW02, Ran13, Ran14]
  - software tools.
Strategy synthesis in stochastic environments

**Strategy** = formal model of how to control the system

1. How complex is it to decide if a winning strategy exists?
2. How complex such a strategy needs to be? **Simpler is better.**
3. Can we synthesize one efficiently?

⇒ Depends on the winning objective, the exact type of interaction, etc.
Aim of this talk

Flavor of $\neq$ types of **useful strategies** in stochastic environments.

- Joint paper\(^1\) with J.-F. Raskin and O. Sankur (ULB) [RRS15b]
- Full paper available on arXiv: abs/1411.0835

---

\(^1\)Invited talk in VMCAI 2015 - 16th International Conference on Verification, Model Checking, and Abstract Interpretation.
Aim of this talk

Flavor of ≠ types of useful strategies in stochastic environments.

- Joint paper\(^1\) with J.-F. Raskin and O. Sankur (ULB) [RRS15b]
- Full paper available on arXiv: abs/1411.0835

Applications to the shortest path problem.

Find a path of minimal length in a weighted graph (Dijkstra, Bellman-Ford, etc) [CGR96].

---

\(^1\)Invited talk in VMCAI 2015 - 16th International Conference on Verification, Model Checking, and Abstract Interpretation.
Aim of this talk

Flavor of different types of useful strategies in stochastic environments.

- Joint paper\(^1\) with J.-F. Raskin and O. Sankur (ULB) [RRS15b]
- Full paper available on arXiv: abs/1411.0835

Applications to the shortest path problem.

What if the environment is uncertain? E.g., in case of heavy traffic, some roads may be crowded.

\(^1\)Invited talk in VMCAI 2015 - 16th International Conference on Verification, Model Checking, and Abstract Interpretation.
Planning a journey in an uncertain environment

Each action takes time, target = work.

▷ What kind of strategies are we looking for when the environment is stochastic (MDP)?
Solution 1: minimize the *expected* time to work

- “Average” performance: meaningful when you journey often.
- **Simple strategies** suffice: no memory, no randomness.
- Taking the **car** is optimal: $\mathbb{E}_D(\text{TS}^{\text{work}}) = 33$. 
Solution 2: traveling without taking too many risks

Minimizing the *expected time* to destination makes sense if we travel often and it is not a problem to be late.

With car, in 10% of the cases, the journey takes 71 minutes.
Solution 2: traveling without taking too many risks

Most bosses will not be happy if we are late too often... What if we are risk-averse and want to avoid that?
Solution 2: maximize the *probability* to be on time

**Specification:** reach work within 40 minutes with 0.95 probability
Solution 2: maximize the *probability* to be on time

**Specification:** reach work within 40 minutes with 0.95 probability

**Sample strategy:** take the train \( \sim \mathbb{P}_D[TS^{work} \leq 40] = 0.99 \)

**Bad choices:** car (0.9) and bike (0.0)
Solution 3: strict worst-case guarantees

Specification: 

**guarantee** that work is reached within 60 minutes (to avoid missing an important meeting)
Solution 3: strict worst-case guarantees

**Specification:** guarantee that work is reached within 60 minutes (to avoid missing an important meeting)

**Sample strategy:** bike $\rightsquigarrow$ worst-case reaching time $= 45$ minutes.

**Bad choices:** train ($wc = \infty$) and car ($wc = 71$)
Solution 3: strict worst-case guarantees

Worst-case analysis $\sim$ two-player game against an antagonistic adversary (bad guy)

▷ forget about probabilities and give the choice of transitions to the adversary
Solution 4: minimize the *expected* time under strict worst-case guarantees

- **Expected time:**  
  - **car** $\sim \mathbb{E} = 33$ but $wc = 71 > 60$
  - **Worst-case:** **bike** $\sim wc = 45 < 60$ but $\mathbb{E} = 45 >>> 33$
Solution 4: minimize the *expected* time under strict worst-case guarantees

In practice, we want both! Can we do better?

▷ **Beyond worst-case synthesis** [BFRR14b, BFRR14a]: minimize the expected time under the worst-case constraint.
Solution 4: minimize the *expected* time under strict worst-case guarantees

Sample strategy: try train up to 3 delays then switch to bike.

\[ wc = 58 < 60 \text{ and } E \approx 37.34 < < 45 \]

\[ \rightarrow \text{ Strategies need } \textbf{memory} \rightarrow \text{ more complex!} \]
Solution 5: multiple objectives $\Rightarrow$ trade-offs

Two-dimensional weights on actions: *time* and *cost*.

Often necessary to consider trade-offs: e.g., between the probability to reach work in due time and the risks of an expensive journey.
Solution 5: multiple objectives $\Rightarrow$ trade-offs

Solution 2 (probability) can only ensure a **single constraint**.

- **C1**: 80% of runs reach work in at most 40 minutes.
  - Taxi $\sim \leq 10$ minutes with probability $0.99 > 0.8$. 
Solution 5: multiple objectives $\Rightarrow$ trade-offs

Solution 2 (probability) can only ensure a **single constraint**.

- **C1**: 80% of runs reach work in at most 40 minutes.
  - Taxi $\sim \leq 10$ minutes with probability $0.99 > 0.8$.
- **C2**: 50% of them cost at most 10$ to reach work.
  - Bus $\sim \geq 70\%$ of the runs reach work for 3$.
Solution 5: multiple objectives $\Rightarrow$ trade-offs

Solution 2 (probability) can only ensure a **single constraint**.

- **C1**: 80% of runs reach work in at most 40 minutes.
  - $\Rightarrow$ Taxi $\sim \leq 10$ minutes with probability $0.99 > 0.8$.

- **C2**: 50% of them cost at most 10$ to reach work.
  - $\Rightarrow$ Bus $\sim \geq 70\%$ of the runs reach work for 3$.

Taxi $\not\models$ C2, bus $\not\models$ C1. What if we want C1 $\land$ C2?
Solution 5: multiple objectives ⇒ trade-offs

- **C1**: 80% of runs reach work in at most 40 minutes.
- **C2**: 50% of them cost at most 10$ to reach work.

Study of **multi-constraint percentile queries** [RRS15a].

- Sample strategy: bus once, then taxi. Requires **memory**.
- Another strategy: bus with probability 3/5, taxi with probability 2/5. Requires **randomness**.
Solution 5: multiple objectives ⇒ trade-offs

- **C1**: 80% of runs reach work in at most 40 minutes.
- **C2**: 50% of them cost at most 10$ to reach work.

Study of *multi-constraint percentile queries* [RRS15a].

In general, *both memory and randomness* are required.

≠ previous problems \(\sim\) more complex!
Our research aims at:

- defining meaningful *strategy concepts*,
- providing *algorithms* and *tools* to compute those strategies,
- classifying the *complexity* of the different problems from a theoretical standpoint.

→ Is it mathematically possible to obtain efficient algorithms?
Algorithmic complexity: hierarchy of problems

- **Solutions 1 (E)** and 3 (wc)
- **Solutions 2 (P)** and 5 (percentile)
- **Solution 4 (BWC)**

- **NP**
- **coNP**
- **NP∩coNP**
- **NP-**
- **PSPACE**
- **EXPTIME**
- **EXPSPACE**
- **LOGSPACE**
- **LOG**
- **PR**
- **ELEMENTARY**
- **P**
- **LOGSPACE**
- **NP**
- **coNP**
- **NP-**

- **UNDECIDABLE**
- not computable by an algorithm
Thank you! Any question?
References

Véronique Bruyère, Emmanuel Filiot, Mickael Randour, and Jean-François Raskin.
Expectations or guarantees? I want it all! A crossroad between games and MDPs.

Véronique Bruyère, Emmanuel Filiot, Mickael Randour, and Jean-François Raskin.
Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games.

Boris V. Cherkassky, Andrew V. Goldberg, and Tomasz Radzik.
Shortest paths algorithms: Theory and experimental evaluation.

E. Grädel, W. Thomas, and T. Wilke, editors.

M. Randour.
Automated synthesis of reliable and efficient systems through game theory: A case study.

M. Randour.
Synthesis in Multi-Criteria Quantitative Games.

Mickael Randour, Jean-François Raskin, and Ocan Sankur.
Percentile queries in multi-dimensional Markov decision processes.
Mickael Randour, Jean-François Raskin, and Ocan Sankur.  
Variations on the stochastic shortest path problem.  

Jean-François Raskin and Ocan Sankur.  
Multiple-environment Markov decision processes.  
### Overview of theoretical results

<table>
<thead>
<tr>
<th>SSP</th>
<th>Complexity</th>
<th>Strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSP-E</td>
<td>PTIME</td>
<td>pure memoryless</td>
</tr>
<tr>
<td>SSP-P</td>
<td>pseudo-PTIME / PSPACE-h.</td>
<td>pure pseudo-poly.</td>
</tr>
<tr>
<td>SSP-G</td>
<td>PTIME</td>
<td>pure memoryless</td>
</tr>
<tr>
<td>SSP-WE</td>
<td>pseudo-PTIME / NP-h.</td>
<td>pure pseudo-poly.</td>
</tr>
<tr>
<td>SSP-PQ</td>
<td>EXPTIME (p.-PTIME) / PSPACE-h.</td>
<td>randomized exponential</td>
</tr>
</tbody>
</table>