Rich Behavioral Models: Illustration on Journey Planning

Mickael Randour
F.R.S.-FNRS & UMONS – Université de Mons, Belgium

April 24, 2019

LSV seminar – ENS Paris-Saclay
The talk in one slide

Strategy synthesis for Markov Decision Processes (MDPs)

Finding **good** controllers for systems interacting with a *stochastic* environment.

- Good? Performance evaluated through *payoff functions*.
- Usual problem is to optimize the *expected performance* or the *probability of achieving a given performance level*.
- Not sufficient for many practical applications.
  - Several extensions, more expressive but also more complex…

Aim of this survey talk

Give a flavor of classical questions and extensions (*rich behavioral models*), illustrated on the stochastic shortest path (SSP).
1. Context, MDPs, strategies

2. Classical stochastic shortest path problems

3. Good expectation under acceptable worst-case

4. Percentile queries in multi-dimensional MDPs

5. Conclusion
1. Context, MDPs, strategies

2. Classical stochastic shortest path problems

3. Good expectation under acceptable worst-case

4. Percentile queries in multi-dimensional MDPs

5. Conclusion
Multi-criteria quantitative synthesis

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

- Model of the (discrete) interaction?
  - Antagonistic environment: 2-player game on graph.
  - **Stochastic environment**: MDP.

- **Quantitative** specifications. Examples:
  - Reach a state \( s \) before \( x \) time units \( \rightarrow \) shortest path.
  - Minimize the average response-time \( \rightarrow \) mean-payoff.

- Focus on **multi-criteria quantitative models**
  - to reason about **trade-offs** and **interplays**.
Strategy (policy) synthesis for MDPs

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?
Markov decision processes

- **MDP** $D = (S, s_{\text{init}}, A, \delta, w)$.
  - Finite sets of states $S$ and actions $A$,
  - probabilistic transition $\delta: S \times A \rightarrow \mathcal{D}(S)$,
  - weight function $w: A \rightarrow \mathbb{Z}$.

- **Run** (or play): $\rho = s_1 a_1 \ldots a_{n-1} s_n \ldots$ such that $\delta(s_i, a_i, s_{i+1}) > 0$ for all $i \geq 1$.
  - Set of runs $\mathcal{R}(D)$.
  - Set of histories (finite runs) $\mathcal{H}(D)$.

- **Strategy** $\sigma: \mathcal{H}(D) \rightarrow \mathcal{D}(A)$.
  - $\forall h$ ending in $s$, $\text{Supp}(\sigma(h)) \in A(s)$. 
Markov decision processes

Sample *pure memoryless* strategy $\sigma$.

Sample run $\rho = s_1 a_1 s_2 a_2 s_1 a_1 s_2 a_2 (s_3 a_3 s_4 a_4) \omega$.

Other possible run $\rho' = s_1 a_1 s_2 a_2 (s_3 a_3 s_4 a_4) \omega$.

- Strategies may use
  - finite or infinite *memory*,
  - *randomness*.

- **Payoff functions** map runs to numerical values:
  - truncated sum up to $T = \{s_3\}$: $TS^T(\rho) = 2$, $TS^T(\rho') = 1$,
  - mean-payoff: $MP(\rho) = MP(\rho') = 1/2$,
  - many more.
Markov chains

Once strategy $\sigma$ fixed, fully stochastic process: $
\leadsto \text{Markov chain (MC) } M.$

State space $=$ product of the MDP and the memory of $\sigma$.

- Event $\mathcal{E} \subseteq \mathcal{R}(M)$
  - probability $\mathbb{P}_M(\mathcal{E})$
- Measurable $f : \mathcal{R}(M) \to \mathbb{R} \cup \{\infty\}$,
  - expected value $\mathbb{E}_M(f)$
Aim of this survey

Compare different types of quantitative specifications for MDPs

- w.r.t. the complexity of the decision problem,
- w.r.t. the complexity of winning strategies.

Recent extensions share a common philosophy: framework for the synthesis of strategies with richer performance guarantees.

- Our work deals with many different payoff functions.

Focus on the shortest path problem in this talk.

- Not the most involved technically, natural applications.
- Useful to understand the practical interest of each variant.

Joint work with R. Berthon, V. Bruyère, E. Filiot, J.-F. Raskin, O. Sankur [BFRR17, RRS17, RRS15, BCH+16, Ran16, BRR17].
1. Context, MDPs, strategies

2. Classical stochastic shortest path problems

3. Good expectation under acceptable worst-case

4. Percentile queries in multi-dimensional MDPs

5. Conclusion
Stochastic shortest path

Shortest path problem for *weighted graphs*

Given state $s \in S$ and target set $T \subseteq S$, find a path from $s$ to a state $t \in T$ that minimizes the sum of weights along edges.

- PTIME algorithms (Dijkstra, Bellman-Ford, etc) [CGR96].

We focus on MDPs with *strictly positive weights* for the SSP.

- **Truncated sum** payoff function for $\rho = s_1a_1s_2a_2 \ldots$ and target set $T$:

$$TS_T(\rho) = \begin{cases} \sum_{j=1}^{n-1} w(a_j) & \text{if } s_n \text{ first visit of } T, \\ \infty & \text{if } T \text{ is never reached.} \end{cases}$$
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?
SSP-E: minimizing the expected length to target

SSP-E problem

Given MDP $D = (S, s_{init}, A, \delta, w)$, target set $T$ and threshold $\ell \in \mathbb{Q}$, decide if there exists $\sigma$ such that $\mathbb{E}_D(\sigma)(TS^T) \leq \ell$.

Theorem [BT91]

The SSP-E problem can be decided in polynomial time. Optimal pure memoryless strategies always exist and can be constructed in polynomial time.
SSP-E: illustration

- Pure memoryless strategies suffice.
- Taking the **car** is optimal: $E_D^\sigma(TS^T) = 33$. 
SSP-E: PTIME algorithm

1. Graph analysis (linear time):
   - s not connected to $T \Rightarrow \infty$ and remove,
   - $s \in T \Rightarrow 0$.

2. Linear programming (LP, polynomial time).

For each $s \in S \setminus T$, one variable $x_s$,

$$\max \sum_{s \in S \setminus T} x_s$$

under the constraints

$$x_s \leq w(a) + \sum_{s' \in S \setminus T} \delta(s, a, s') \cdot x_{s'}$$

for all $s \in S \setminus T$, for all $a \in A(s)$. 
SSP-E: PTIME algorithm

1. **Graph analysis (linear time):**
   - $s$ not connected to $T$ $\Rightarrow \infty$ and remove,
   - $s \in T$ $\Rightarrow 0$.

2. **Linear programming (LP, polynomial time).**

    Optimal solution $\mathbf{v}$:
    \[ \mathbf{v}_s = \text{expectation from } s \text{ to } T \text{ under an optimal strategy}. \]

    Optimal pure memoryless strategy $\sigma^\mathbf{v}$:
    \[
    \sigma^\mathbf{v}(s) = \arg \min_{a \in A(s)} \left[ w(a) + \sum_{s' \in S \setminus T} \delta(s, a, s') \cdot \mathbf{v}_{s'} \right].
    \]
    \[ \sim \text{ Playing optimally } = \text{ locally optimizing present } + \text{ future}. \]
SSP-E: PTIME algorithm

1. Graph analysis (linear time):
   - $s$ not connected to $T \Rightarrow \infty$ and remove,
   - $s \in T \Rightarrow 0$.

2. Linear programming (LP, polynomial time).

In practice, **value and strategy iteration** algorithms often used:

- best performance in most cases but **exponential** in the worst-case,
- fixed point algorithms, successive solution improvements [BT91, dA99, HM14].
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.
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?
SSP-P: forcing short paths with high probability

SSP-P problem

Given MDP $D = (S, s_{\text{init}}, A, \delta, w)$, target set $T$, threshold $\ell \in \mathbb{N}$, and probability threshold $\alpha \in [0, 1] \cap \mathbb{Q}$, decide if there exists a strategy $\sigma$ such that $\mathbb{P}_D^\sigma \left[ \{ \rho \in \mathcal{R}_{s_{\text{init}}}(D) \mid TS_T^\sigma(\rho) \leq \ell \} \right] \geq \alpha$.

Theorem

The SSP-P problem can be decided in pseudo-polynomial time, and it is PSPACE-hard. Optimal pure strategies with pseudo-polynomial memory always exist and can be constructed in pseudo-polynomial time.

See [HK15] for hardness and for example [RRS17] for algorithm.
SSP-P: illustration

**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)
SSP-P: pseudo-PTIME algorithm (1/2)

Key idea: pseudo-PTIME reduction to the **stochastic reachability problem** (SR)

**SR problem**

Given unweighted MDP $D = (S, s_{\text{init}}, A, \delta)$, target set $T$ and probability threshold $\alpha \in [0, 1] \cap \mathbb{Q}$, decide if there exists a strategy $\sigma$ such that $P_D^{\sigma}[\diamond T] \geq \alpha$.

**Theorem**

The SR problem can be decided in polynomial time. Optimal pure memoryless strategies always exist and can be constructed in polynomial time.

▷ Linear programming (similar to SSP-E).
SSP-P: pseudo-PTIME algorithm (2/2)

**Sketch of the reduction:**

1. Start from $D$, $T = \{s_2\}$, and $\ell = 7$.

2. Build $D_\ell$ by unfolding $D$, tracking the current sum *up to the threshold* $\ell$, and integrating it in the states of the expanded MDP.
SSP-P: pseudo-PTIME algorithm (2/2)
SSP-P: pseudo-PTIME algorithm (2/2)

3 Relation between runs of $D$ and $D_\ell$:

$$\text{TS}^T(\rho) \leq \ell \iff \rho' \models \Diamond T', \ T' = T \times \{0, 1, \ldots, \ell\}.$$ 

4 Solve the SR problem on $D_\ell$.

- Memoryless strategy in $D_\ell \leadsto$ pseudo-polynomial memory in $D$ in general.
SSP-P: pseudo-PTIME algorithm (2/2)

If we just want to minimize the risk of exceeding \( \ell = 7 \),

- an obvious possibility is to play \( b \) directly,
- playing \( a \) only once is also acceptable.

For the SSP-P problem, **both strategies are equivalent**.

\( \Rightarrow \) We need richer models to discriminate them!


**Related work (non-exhaustive)**

- SSP-P problem with relaxed hypotheses [Oht04, SO13].
- SSP-E problem with relaxed hypotheses [BBD$^+18$].
- *Quantile queries* [UB13]: minimizing the value $\ell$ of an SSP-P problem for some fixed $\alpha$. Extended to *cost problems* [HK15, HKL17].
- SSP-E problem in **multi-dimensional** MDPs [FKN$^+11$].
1. Context, MDPs, strategies

2. Classical stochastic shortest path problems

3. **Good expectation under acceptable worst-case**

4. Percentile queries in multi-dimensional MDPs

5. Conclusion
SP-G: strict worst-case guarantees

**Specification:** *guarantee* that work is reached within 60 minutes (to avoid missing an important meeting).
SP-G: strict worst-case guarantees

Winning surely (worst-case) $\neq$ almost-surely (proba. 1).

- Train ensures reaching work with probability one, but does not prevent runs where work is never reached.
SP-G: strict worst-case guarantees

Worst-case analysis $\leadsto$ two-player game against an antagonistic adversary.

- Forget about probabilities and give the choice of transitions to the adversary.
**SP-G: shortest path game problem**

**SP-G problem**

Given MDP $D = (S, s_{init}, A, \delta, w)$, target set $T$ and threshold $\ell \in \mathbb{N}$, decide if there exists a strategy $\sigma$ such that for all $\rho \in \text{Out}_D^\sigma$, we have that $TS^T(\rho) \leq \ell$.

**Theorem [KBB+08]**

The SP-G problem can be decided in polynomial time. Optimal pure memoryless strategies always exist and can be constructed in polynomial time.

▶ Dynamic programming.
Related work (non-exhaustive)

- Pseudo-PTIME for arbitrary weights [BGHM17, FGR15].

- Arbitrary weights + multiple dimensions \(\not\sim\) undecidable (by adapting the proof of [CDRR15] for total-payoff).
SSP-WE = SP-G \cap SSP-E - illustration

- **SSP-E**: car $\sim E = 33$ but $wc = 71 > 60$
- **SP-G**: bike $\sim wc = 45 < 60$ but $E = 45 >>> 33$
SSP-WE = SP-G ∩ SSP-E - illustration

Can we do better?

▸ **Beyond worst-case synthesis** [BFRR17]: minimize the expected time under the worst-case constraint.
SSP-WE = SP-G ∩ SSP-E - illustration

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

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

\[ \implies \text{pure finite-memory strategy} \]
SSP-WE: beyond worst-case synthesis

**SSP-WE problem**

Given MDP $D = (S, s_{\text{init}}, A, \delta, w)$, target set $T$, and thresholds $\ell_1 \in \mathbb{N}$, $\ell_2 \in \mathbb{Q}$, decide if there exists a strategy $\sigma$ such that:

1. $\forall \rho \in \text{Out}_D^\sigma: \text{TS}_T^T(\rho) \leq \ell_1$,
2. $\mathbb{E}_D^\sigma(\text{TS}_T^T) \leq \ell_2$.

**Theorem [BFRR17]**

The SSP-WE problem can be decided in pseudo-polynomial time and is NP-hard. Pure pseudo-polynomial-memory strategies are always sufficient and in general necessary, and satisfying strategies can be constructed in pseudo-polynomial time.
SSP-WE: pseudo-PTIME algorithm

Consider SSP-WE problem for $\ell_1 = 7$ (wc), $\ell_2 = 4.8$ (E).

- Reduction to the SSP-E problem on a pseudo-polynomial-size expanded MDP.

1. Build unfolding as for SSP-P problem w.r.t. worst-case threshold $\ell_1$. 
SSP-WE: pseudo-PTIME algorithm

Here, $E_\sigma D'$ for $TS'_T = 9/2$. 

Rich Behavioral Models  Mickael Randour  27 / 41
SSP-WE: pseudo-PTIME algorithm

2 Compute $R$, the attractor of $T' = T \times \{0, 1, \ldots, \ell_1\}$.

3 Restrict MDP to $D' = D_{\ell_1} |_R$, the safe part w.r.t. SP-G.
SSP-WE: pseudo-PTIME algorithm

2. Compute $R$, the attractor of $T' = T \times \{0, 1, \ldots, \ell_1\}$.
3. Restrict MDP to $D' = D_{\ell_1} \downarrow R$, the safe part w.r.t. SP-G.
SSP-WE: pseudo-PTIME algorithm

4. Compute memoryless optimal strategy $\sigma$ in $D'$ for SSP-E.
5. Answer is YES iff $E_{D'}(TS^{T'}) \leq \ell_2$.

Here,

$$E_{D'}(TS^{T'}) = \frac{9}{2}.$$
# SSP-WE: wrap-up

<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>
</tbody>
</table>

- NP-hardness $\Rightarrow$ inherently harder than SSP-E and SSP-G.
Related work (non-exhaustive)

- BWC synthesis problems for mean-payoff \([\text{BFRR17}]\) and parity \([\text{BRR17}]\) belong to \(\text{NP} \cap \text{coNP}\). Much more involved technically.
  
  \[\implies\] Additional modeling power for free w.r.t. worst-case problems.

- Multi-dimensional extension for mean-payoff \([\text{CR15}]\).
- Integration of BWC concepts in \text{Uppaal} \([\text{DJL}^+14]\).
- Optimizing the expected mean-payoff under energy constraints \([\text{BKN16}]\) or Boolean constraints \([\text{AKV16}]\).
- Recent extensions to POMDPs \([\text{CNP}^+17, \text{KPR18, CENR18}]\).
- Conditional value-at-risk \([\text{KM18}]\).
1. Context, MDPs, strategies

2. Classical stochastic shortest path problems

3. Good expectation under acceptable worst-case

4. Percentile queries in multi-dimensional MDPs

5. Conclusion
Multiple objectives $\implies$ 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.
Multiple objectives $\implies$ trade-offs

SSP-P problem considers a **single percentile 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$.

Taxi $\not\equiv$ C2, bus $\not\equiv$ C1. What if we want C1 $\land$ C2?
Multiple objectives $\implies$ 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** [RRS17].

- Sample strategy: bus once, then taxi. Requires *memory*.
- Another strategy: bus with probability 3/5, taxi with probability 2/5. Requires *randomness*.
Multiple objectives $\Rightarrow$ 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 [RRS17].

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

$\neq$ Previous problems.
SSP-PQ: multi-constraint percentile queries (1/2)

SSP-PQ problem

Given \(d\)-dimensional MDP \(D = (S, s_{init}, A, \delta, w)\), and \(q \in \mathbb{N}\) percentile constraints described by target sets \(T_i \subseteq S\), dimensions \(k_i \in \{1, \ldots, d\}\), value thresholds \(\ell_i \in \mathbb{N}\) and probability thresholds \(\alpha_i \in [0, 1] \cap \mathbb{Q}\), where \(i \in \{1, \ldots, q\}\), decide if there exists a strategy \(\sigma\) such that query \(Q\) holds, with

\[
Q := \bigwedge_{i=1}^{q} \mathbb{P}^\sigma_D[TS_{T_i}^{T_i} \leq \ell_i] \geq \alpha_i,
\]

where \(TS_{T_i}^{T_i}\) denotes the truncated sum on dimension \(k_i\) and w.r.t. target set \(T_i\).

Very general framework: multiple constraints related to \(\neq\) dimensions, and \(\neq\) target sets \(\implies\) great flexibility in modeling.
SSP-PQ: multi-constraint percentile queries (2/2)

Theorem [RRS17]

The SSP-PQ problem can be decided in
- **exponential time** in general,
- **pseudo-polynomial time** for single-dimension single-target multi-constraint queries.

It is **PSPACE-hard** even for single-constraint queries. **Randomized exponential-memory** strategies are always sufficient and in general necessary, and satisfying strategies can be constructed in exponential time.

- Unfolding + multiple reachability problem [EKVY08, RRS17].
- PSPACE-hardness already true for SSP-P [HK15].
- SSP-PQ = wide extension for **basically no price in complexity**.
### SSP-PQ: wrap-up

<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>

- SSP-PQ is undecidable for arbitrary weights in multi-dimensional MDPs, even with a unique target set [RRS17].

- Clever unfolding technique in [HJKQ18].
Percentile queries: overview (1/2)

- **Wide range of payoff functions**
  - multiple reachability,
  - mean-payoff (\(\overline{\text{MP}}, \text{MP}\)),
  - discounted sum (DS).

- **Several variants:**
  - multi-dim. multi-constraint,
  - single-constraint.

- For each one:
  - algorithms,
  - memory requirements.

- **Complete picture** for this new framework.
Percentile queries: overview (2/2)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reachability</td>
<td>P [Put94]</td>
<td>P(D) \cdot E(Q) [EKVY08], PSPACE-h</td>
<td>—</td>
</tr>
<tr>
<td>$f \in \mathcal{F}$</td>
<td>P [CH09]</td>
<td>P</td>
<td>P(D) \cdot E(Q) PSPACE-h.</td>
</tr>
<tr>
<td>MP</td>
<td>P [Put94]</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>MP</td>
<td>P [Put94]</td>
<td>P(D) \cdot E(Q)</td>
<td>P(D) \cdot E(Q)</td>
</tr>
<tr>
<td>SP</td>
<td>P(D) \cdot P_{ps}(Q) [HK15] PSPACE-h. [HK15]</td>
<td>P(D) \cdot P_{ps}(Q) (one target) PSPACE-h. [HK15]</td>
<td>P(D) \cdot E(Q) PSPACE-h. [HK15]</td>
</tr>
<tr>
<td>$\varepsilon$-gap DS</td>
<td>P_{ps}(D, Q, \varepsilon) NP-h.</td>
<td>P_{ps}(D, \varepsilon) \cdot E(Q) NP-h.</td>
<td>P_{ps}(D, \varepsilon) \cdot E(Q) PSPACE-h.</td>
</tr>
</tbody>
</table>

$\mathcal{F} = \{ \inf, \sup, \lim \inf, \lim \sup \}$

$D = $ model size, $Q = $ query size

$P(x)$, $E(x)$ and $P_{ps}(x)$ resp. denote polynomial, exponential and pseudo-polynomial time in parameter $x$.

All results without reference are established in [RRS17].
Percentile queries: overview (2/2)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reachability</td>
<td>P [Put94]</td>
<td>P(D)·E(Q) [EKVY08], PSPACE-h</td>
<td>—</td>
</tr>
<tr>
<td>$f \in \mathcal{F}$</td>
<td>P [CH09]</td>
<td>P</td>
<td>P(D)·E(Q) PSPACE-h.</td>
</tr>
<tr>
<td>MP</td>
<td>P [Put94]</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>MP</td>
<td>P [Put94]</td>
<td>P(D)·E(Q)</td>
<td>P(D)·E(Q)</td>
</tr>
<tr>
<td>SP</td>
<td>P(D)·P_{ps}(Q) [HK15]</td>
<td>P(D)·P_{ps}(Q) (one target)</td>
<td>P(D)·E(Q) PSPACE-h. [HK15]</td>
</tr>
<tr>
<td>$\varepsilon$-gap DS</td>
<td>$P_{ps}(D, Q, \varepsilon)$</td>
<td>$P_{ps}(D, \varepsilon)·E(Q)$</td>
<td>$P_{ps}(D, \varepsilon)·E(Q)$</td>
</tr>
</tbody>
</table>

In most cases, only **polynomial in the model size**.

▷ In practice, the query size can often be bounded while the model can be very large.
Related work (non-exhaustive)

- Percentile + expected value for shortest path [BGMR18].
- Multi-dimensional quantiles [HKL17].
1  Context, MDPs, strategies

2  Classical stochastic shortest path problems

3  Good expectation under acceptable worst-case

4  Percentile queries in multi-dimensional MDPs

5  Conclusion
Summary: stochastic shortest path problem

- **SSP-E**: minimize the expected sum to target.
  - Actual outcomes may vary greatly.

- **SSP-P**: maximize the probability of acceptable performance.
  - No control over the quality of bad runs, no average-case performance.

- **SP-G**: maximize the worst-case performance, extreme risk-aversion.
  - Strict worst-case guarantees, no average-case performance.

- **SSP-WE**: SSP-E $\cap$ SP-G.
  - Based on beyond worst-case synthesis [BFRR17].

- **SSP-PQ**: extends SSP-P to multi-constraint percentile queries [RRS17].
  - Multi-dimensional, flexible, trade-offs.
  - Complexity usually acceptable w.r.t. model size.
Rich behavioral models: challenges

1. **Plethora of theoretical models.**
   - Fundamental question: identify and understand the common core, advance toward unification.
   - Can be an obstacle to adoption by practitioners.

2. **Practical applicability.**
   - Efficiency must be increased (e.g., by using learning techniques).
   - Tool support is key.
If you are interested... 

... consider attending MoRe 2019, the 2nd International Workshop on Multi-objective Reasoning in Verification and Synthesis, to be held in Vancouver (LICS 2019), on June 22.

Thank you! Any question?
Shaull Almagor, Orna Kupferman, and Yaron Velner.
Minimizing expected cost under hard boolean constraints, with applications to quantitative synthesis.

Christel Baier, Nathalie Bertrand, Clemens Dubslaff, Daniel Gburek, and Ocan Sankur.
Stochastic shortest paths and weight-bounded properties in Markov decision processes.
In Dawar and Grädel [DG18], pages 86–94.

Romain Brenguier, Lorenzo Clemente, Paul Hunter, Guillermo A. Pérez, Mickael Randour, Jean-François Raskin, Ocan Sankur, and Mathieu Sassolas.
Non-zero sum games for reactive synthesis.

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

Thomas Brihaye, Gilles Geeraerts, Axel Haddad, and Benjamin Monmege.
Pseudopolynomial iterative algorithm to solve total-payoff games and min-cost reachability games.
References II

Patricia Bouyer, Mauricio González, Nicolas Markey, and Mickael Randour.
Multi-weighted markov decision processes with reachability objectives.

Tomás Brázdil, Antonín Kucera, and Petr Novotný.
Optimizing the expected mean payoff in energy Markov decision processes.

Raphaël Berthon, Mickael Randour, and Jean-François Raskin.
Threshold constraints with guarantees for parity objectives in Markov decision processes.

Dimitri P. Bertsekas and John N. Tsitsiklis.
An analysis of stochastic shortest path problems.

Krishnendu Chatterjee, Laurent Doyen, Mickael Randour, and Jean-François Raskin.
Looking at mean-payoff and total-payoff through windows.
References III

Krishnendu Chatterjee, Adrián Elgyütt, Petr Novotný, and Owen Rouillé.
Expectation optimization with probabilistic guarantees in POMDPs with discounted-sum objectives.

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

Krishnendu Chatterjee and Thomas A. Henzinger.
Probabilistic systems with limsup and liminf objectives.

Krishnendu Chatterjee, Petr Novotný, Guillermo A. Pérez, Jean-François Raskin, and Dorde Zikelic.
Optimizing expectation with guarantees in POMDPs.

Lorenzo Clemente and Jean-François Raskin.
Multidimensional beyond worst-case and almost-sure problems for mean-payoff objectives.
 Luca de Alfaro.  
Computing minimum and maximum reachability times in probabilistic systems.  

 Anuj Dawar and Erich Grädel, editors.  

 Alexandre David, Peter Gjøl Jensen, Kim Guldstrand Larsen, Axel Legay, Didier Lime, Mathias Grund Sørensen, and Jakob Haahr Taankvist.  
On time with minimal expected cost!  

 Kousha Etessami, Marta Z. Kwiatkowska, Moshe Y. Vardi, and Mihalis Yannakakis.  
Multi-objective model checking of Markov decision processes.  

 Emmanuel Filiot, Raffaella Gentilini, and Jean-François Raskin.  
Quantitative languages defined by functional automata.  
References V

Vojtech Forejt, Marta Z. Kwiatkowska, Gethin Norman, David Parker, and Hongyang Qu.
Quantitative multi-objective verification for probabilistic systems.

Arnd Hartmanns, Sebastian Junges, Joost-Pieter Katoen, and Tim Quatmann.
Multi-cost bounded reachability in MDP.

Christoph Haase and Stefan Kiefer.
The odds of staying on budget.

Christoph Haase, Stefan Kiefer, and Markus Lohrey.
Computing quantiles in Markov chains with multi-dimensional costs.
References VI

Serge Haddad and Benjamin Monmege.
Reachability in MDPs: Refining convergence of value iteration.

Leonid Khachiyan, Endre Boros, Konrad Borys, Khaled M. Elbassioni, Vladimir Gurvich, Gábor Rudolf, and Jihui Zhao.
On short paths interdiction problems: Total and node-wise limited interdiction.

Jan Křížek and Tobias Meggendorfer.
Conditional value-at-risk for reachability and mean payoff in Markov decision processes.
In Dawar and Grädel [DG18], pages 609–618.

Jan Křížek, Guillermo A. Pérez, and Jean-François Raskin.
Learning-based mean-payoff optimization in an unknown MDP under omega-regular constraints.

Yoshio Ohtsubo.
Optimal threshold probability in undiscounted Markov decision processes with a target set.
References VII

Martin L. Puterman.

Mickael Randour.
Reconciling rationality and stochasticity: Rich behavioral models in two-player games.
GAMES 2016, the 5th World Congress of the Game Theory Society, Maastricht, Netherlands.

Mickael Randour, Jean-François Raskin, and Ocan Sankur.
Variations on the stochastic shortest path problem.

Mickael Randour, Jean-François Raskin, and Ocan Sankur.
Percentile queries in multi-dimensional Markov decision processes.

Masahiko Sakaguchi and Yoshio Ohtsubo.
Markov decision processes associated with two threshold probability criteria.
Michael Ummels and Christel Baier.
Computing quantiles in Markov reward models.
SP-G: PTIME algorithm

1. Cycles are bad $\implies$ must reach target within $n = |S|$ steps.

2. $\forall s \in S, \forall i, 0 \leq i \leq n$, compute $C(s, i)$.
   - Lowest bound on cost to $T$ from $s$ that we can ensure in $i$ steps.
   - Dynamic programming (polynomial time).

Initialize

$$\forall s \in T, C(s, 0) = 0, \quad \forall s \in S \setminus T, C(s, 0) = \infty.$$ 

Then, $\forall s \in S, \forall i, 1 \leq i \leq n,$

$$C(s, i) = \min \left[ C(s, i-1), \min_{a \in A(s)} \max_{s' \in \text{Supp}(\delta(s, a))} w(a) + C(s', i-1) \right].$$

3. Winning strategy iff $C(s_{\text{init}}, n) \leq \ell.$
SSP-PQ: EXPTIME / pseudo-PTIME algorithm

1. Build an unfolded MDP $D_\ell$ similar to SSP-P case:
   - stop unfolding when all dimensions reach sum $\ell = \max_i \ell_i$.

2. Maintain single-exponential size by defining an equivalence relation between states of $D_\ell$:
   - $S_\ell \subseteq S \times (\{0, \ldots, \ell\} \cup \{\bot\})^d$,
   - pseudo-poly. if $d = 1$.

3. For each constraint $i$, compute a target set $R_i$ in $D_\ell$:
   - $\rho \models$ constraint $i$ in $D \iff \rho' \models \Diamond R_i$ in $D_\ell$.

4. Solve a multiple reachability problem on $D_\ell$.
   - Generalizes the SR problem [EKVY08, RRS17].
   - Time polynomial in $|D_\ell|$ but exponential in $q$.
   - Single-dim. single target queries $\Rightarrow$ absorbing targets $\Rightarrow$ polynomial-time algorithm.