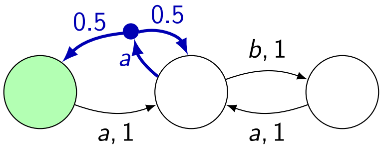
Combining Guaranteed and Stochastic Objectives in Markov Decision Processes

Raphaël Berthon[1,2]
work with: Shibashis Guha [1] Jean-François Raskin [1]

[1]Université libre de Bruxelles, Belgium [2]University of Antwerp, Belgium

September 15, 2020

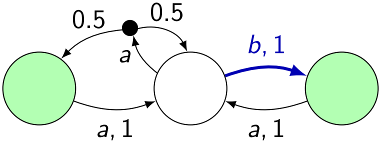
Almost-Sure Objective



$\texttt{AS}(p)$ *Almost-surely* : The objective holds with probability 1.

By only playing action a, we have probability 1 of visiting a green state infinitely often.

Sure Objective



$\mathtt{A}(p)$ *Surely* : The objective holds in all paths. This is the same as having a two-player game with this objective.

By always playing action b, we are sure of visiting a green state infinitely often.

Our Problem

Finding a strategy for Boolean combinations of Sure and Almost-Sure objectives.

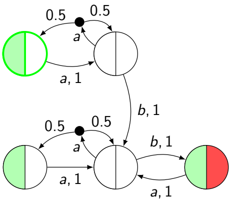
$$\begin{aligned} \mathit{atom} &= \mathtt{A}(\mathtt{p}) \mid \mathtt{E}(\mathtt{p}) \mid \mathtt{AS}(\mathtt{p}) \mid \mathtt{NZ}(\mathtt{p})\ (\mathtt{p} \in \mathtt{parity}) \\ \varphi &= \mathit{atom} \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \neg\varphi \end{aligned}$$

Example of a formula

We can specify that we want the system to behave such that

- it *fulfills a task* with probability greater than 0
- it has probability 1 of *not crashing*
- if there exists a possibility that the system *crashes* then the system must always be able to *raise an alert*

$$\mathtt{NZ}(\mathit{task}) \land \mathtt{AS}(\overline{\mathit{crash}}) \land (\mathtt{E}(\mathit{crash}) \rightarrow \mathtt{A}(\mathit{able\_alert}))$$



General idea of the decision procedure

We guess a conjunction satisfiable on the MDP:

$$\texttt{NZ}(\textit{task}) \land \texttt{AS}(\overline{\textit{crash}}) \land \texttt{A}(\textit{able\_alert})$$

General idea of the decision procedure

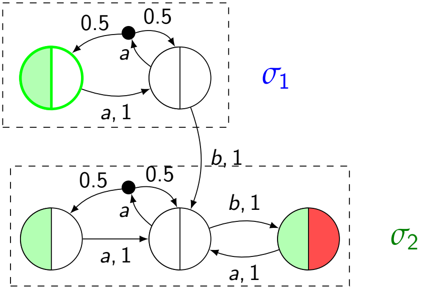
We split the conjunction into relevant sub-formulas:

$$\mathtt{NZ}(\mathit{task}) \land \mathtt{AS}(\overline{\mathit{crash}}) \land \mathtt{A}(\mathit{able\_alert})$$

$$\mathtt{AS}(\overline{\mathit{crash}}) \land \mathtt{A}(\mathit{able\_alert})$$

General idea of the decision procedure

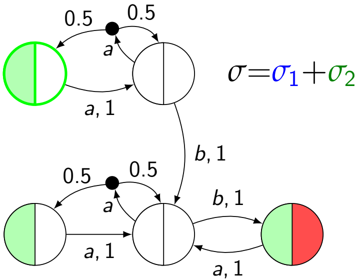
We find the maximal end-component of the MDP with strategies satisfying the sub-formulas:



General idea of the decision procedure

We merge these strategies in a strategy satisfying the initial Boolean formula:

$$\mathtt{NZ}(task) \land \mathtt{AS}(\overline{crash}) \land (\mathtt{E}(crash) \to \mathtt{A}(able\_alert))$$



Our results

We synthesize strategies for Boolean combinations of Sure, Almost-Sure, Non-Zero and Existential objectives. Randomized infinite-memory strategies are needed.

	Membership
$\bigwedge(\mathtt{AS}, \mathtt{NZ}, \mathtt{E})$	P
$\bigwedge(\mathtt{1A}, \mathtt{AS}, \mathtt{NZ}, \mathtt{E})$	$\mathsf{NP} \cap \mathsf{coNP}$
$\mathbb{B}(\mathtt{A}, \mathtt{AS}, \mathtt{NZ}, \mathtt{E})$	$\mathsf{NP}^{\mathsf{NP}} (= \Sigma_2^{\mathsf{P}})$

Future works

- Quantitative objectives enforced with probability greater than a threshold $P_{>k}$.
- Other types of conditions: Streett, Rabin, Muller.

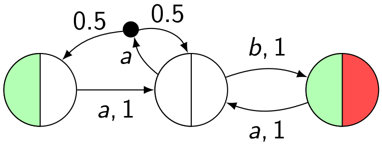
Almagor, S., Kupferman, O., and Velner, Y. (2016).
Minimizing expected cost under hard boolean constraints, with applications to quantitative synthesis.
In Desharnais, J. and Jagadeesan, R., editors, *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, volume 59 of *LIPIcs*, pages 9:1–9:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

B., R., Randour, M., and Raskin, J.-F. (2017).
Threshold constraints with guarantees for parity objectives in markov decision processes.
In *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Bruyere, V., Filiot, E., Randour, M., and Raskin, J.-F. (2017).
Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games.
Information and computation, 254:259–295.

Chatterjee, K. and Piterman, N. (2019).
Combinations of qualitative winning for stochastic parity games.
In *30th International Conference on Concurrency Theory*, page 1.

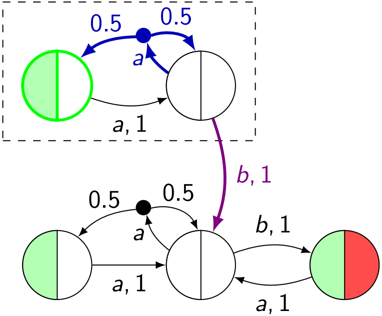
Etessami, K., Kwiatkowska, M., Vardi, M. Y., and Yannakakis, M. (2007).
Multi-objective model checking of markov decision processes.
In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 50–65. Springer.



Type II end-component

We define Type II end-component in such a way that there always exists a strategy staying in the end-component and enforcing a conjunction of sure and almost-sure condition, more formally a formula of the form:

$$\bigwedge_{a \in \mathcal{A}} \mathtt{A}(\mathtt{p}_a) \wedge \bigwedge_{as \in \mathcal{AS}} \mathtt{AS}(\mathtt{p}_{as}).$$



Type III end-component

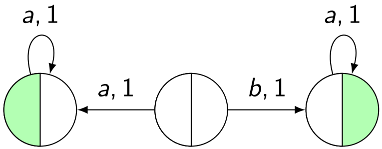
A Type III end-component is characterized by having both:

- A strategy staying in the end-component and satisfying almost-surely all relevant parity conditions.
- Another strategy that can reach a Type II end-component.

These two strategies can be combined to enforce:

$$\bigwedge_{a \in \mathcal{A}} \mathtt{A}(\mathtt{p}_a) \wedge \bigwedge_{as \in \mathcal{AS}} \mathtt{AS}(\mathtt{p}_{as}) \wedge \bigwedge_{nz \in \mathcal{NZ}} \mathtt{NZ}(\mathtt{p}_{nz}).$$

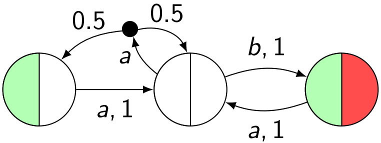
Randomized strategies needed



Combining two non-zero

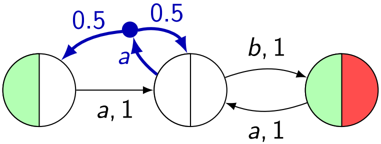
Having non-zero probability of visiting a green right state and a non-zero probability of visiting a green left state uses *randomized strategies*:
Flip a coin. If it gives heads, play *a*, if tails play *b*.

Infinite memory needed



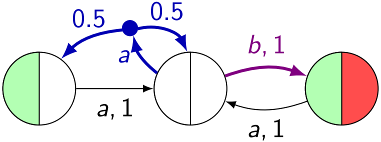
We give a strategy to surely visit infinitely often a green left state and almost surely visiting finitely often a red right state.

Infinite memory needed



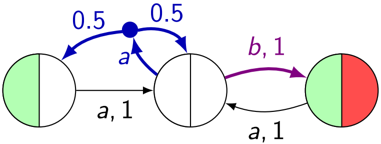
We give a strategy to surely visit infinitely often a green left state and almost surely visiting finitely often a red right state.
Proceed by rounds: at round i, play a^i.

Infinite memory needed



We give a strategy to surely visit infinitely often a green left state and almost surely visiting finitely often a red right state.
Proceed by rounds: at round i, play a^i. If state  is not reached during a round, play b once.

Infinite memory needed



We give a strategy to surely visit infinitely often a green left state and almost surely visiting finitely often a red right state.
Proceed by rounds: at round i, play a^i. If state  is not reached during a round, play b once.
It is clear that in every round a green left state is visited. We can prove there is probability 0 of taking infinitely often action b: we almost-surely visit finitely often the red right state.

Sketch of the general synthesis algorithm

Given a boolean QPL formula φ:
Guess a conjunction $\bigwedge_{a\in\mathcal{A}} \mathtt{A}(\mathsf{p}_a) \wedge \bigwedge_{as\in\mathcal{AS}} \mathtt{AS}(\mathsf{p}_{as}) \wedge \bigwedge_{nz\in\mathcal{NZ}} \mathtt{NZ}(\mathsf{p}_{nz}) \wedge \bigwedge_{e\in\mathcal{E}} \mathtt{E}(\mathsf{p}_e)$

- Prune all states that do not satisfy $\bigwedge_{a\in\mathcal{A}} \mathtt{A}(\mathsf{p}_a) \wedge \bigwedge_{as\in\mathcal{AS}} \mathtt{AS}(\mathsf{p}_{as})$.
  (Type II EC)
- Find strategies σ_i:
  to enforce all $\bigwedge_{a\in\mathcal{A}} \mathtt{A}(\mathsf{p}_a) \wedge \bigwedge_{as\in\mathcal{AS}} \mathtt{AS}(\mathsf{p}_{as}) \wedge \mathtt{NZ}(\mathsf{p}_i)$ (Type III)
  and all $\bigwedge_{a\in\mathcal{A}} \mathtt{A}(\mathsf{p}_a) \wedge \bigwedge_{as\in\mathcal{AS}} \mathtt{AS}(\mathsf{p}_{as}) \wedge \mathtt{E}(\mathsf{p}_i)$.
- Return strategy σ that chooses randomly an i and plays σ_i.

Technical tools

Most of the work is done by finding sets of maximal end-components verifying some properties characterizing a given conjunction of objectives.