

Expectations or Guarantees? I Want It All!

A crossroad between games and MDPs*

Véronique Bruyère

Université de Mons
Belgium

Emmanuel Filiot

Université Libre de Bruxelles
Belgium

Mickael Randour

Université de Mons
Belgium

Jean-François Raskin

Université Libre de Bruxelles
Belgium

When reasoning about the strategic capabilities of an agent, it is important to consider the nature of its adversaries. In the particular context of controller synthesis for quantitative specifications, the usual problem is to devise a strategy for a reactive system which yields some desired performance, taking into account the possible impact of the environment of the system. There are at least two ways to look at this environment. In the classical analysis of two-player quantitative games, the environment is purely antagonistic and the problem is to provide strict performance guarantees. In Markov decision processes, the environment is seen as purely stochastic: the aim is then to optimize the expected payoff, with no guarantee on individual outcomes.

In this expository work, we report on recent results [7, 6] introducing the beyond worst-case synthesis problem, which is to construct strategies that guarantee some quantitative requirement in the worst-case while providing an higher expected value against a particular stochastic model of the environment given as input. This problem is relevant to produce system controllers that provide nice expected performance in the everyday situation while ensuring a strict (but relaxed) performance threshold even in the event of very bad (while unlikely) circumstances. It has been studied for both the mean-payoff and the shortest path quantitative measures.

1 Introduction

Classical models. Two-player zero-sum quantitative games [14, 27, 5] and Markov decision processes (MDPs) [23, 8] are two popular formalisms for modeling decision making in adversarial and uncertain environments respectively. In the former, two players compete with opposite goals (zero-sum), and we want strategies for player 1 (the system) that ensure a given *minimal performance against all possible strategies* of player 2 (its environment). In the latter, the system plays against a stochastic model of its environment, and we want strategies that ensure a *good expected overall performance*. Those two models are well studied and simple optimal memoryless strategies exist for classical objectives such as mean-payoff [21, 14, 15] or shortest path [3, 2]. But both models have clear weaknesses: strategies that are good for the worst-case may exhibit suboptimal behaviors in probable situations while strategies that are good for the expectation may be terrible in some unlikely but possible situations.

What if we want both? In practice, we want strategies that both ensure (a) some worst-case threshold no matter how the adversary behaves (i.e., against any arbitrary strategy) and (b) a good expectation against

*Work partially supported by European project CASSTING (FP7-ICT-601148). Filiot and Randour are respectively F.R.S.-FNRS research associate and research fellow. Raskin is supported by ERC Starting Grant inVEST (279499).

the expected behavior of the adversary (given as a stochastic model). We study how to construct such finite-memory strategies. We consider finite memory for player 1 as it can be implemented in practice (as opposed to infinite memory). Player 2 is not restricted in his choice of strategies, but we show that simple strategies suffice. Our problem, the **beyond worst-case synthesis problem**, makes sense for any quantitative measure. We focus on two classical ones: the *mean-payoff*, and the *shortest path*. Our results are summarized in Table 1.

		worst-case	expected value	BWC
mean-payoff	complexity	$\text{NP} \cap \text{coNP}$	P	$\text{NP} \cap \text{coNP}$
	memory	memoryless		pseudo-poly.
shortest path	complexity	P		pseudo-poly. / NP-hard
	memory	memoryless		pseudo-poly.

Table 1: Overview of decision problem complexities and memory requirements for winning strategies of the first player in games (worst-case), MDPs (expected value) and the BWC setting (combination).

Example. Consider the weighted game in Fig. 1 to illustrate the *shortest path* context. Circle states belong to player 1, square states to player 2, integer labels are durations in minutes, and fractions are probabilities that model the expected behavior of player 2. Player 1 wants a strategy to go from “home” to “work” such that “work” is *guaranteed* to be reached within 60 minutes (to avoid missing an important meeting), and player 1 would also like to minimize the expected time to reach “work”. The strategy that minimizes the expectation is to take the car (expectation is 33 minutes) but it is excluded as there is a possibility to arrive after 60 minutes (in case of heavy traffic). Bicycle is safe but the expectation of this solution is 45 minutes. We can do better with the following strategy: try to take the train, if the train is delayed three time consecutively, then go back home and take the bicycle. This strategy is safe as it always reaches “work” within 59 minutes and its expectation is $\approx 37,56$ minutes (so better than taking directly the bicycle). Observe that this simple example already shows that, unlike the situation for classical games and MDPs, strategies using memory are strictly more powerful than memoryless ones. Our algorithms are able to decide the existence of (and synthesize) such finite-memory strategies.

Related work. This paper gives an expository presentation of results appeared in [7] (an extended version of the paper can be found in [6]). Our problems generalize the corresponding problems for two-player zero-sum games and MDPs. In mean-payoff games, optimal memoryless worst-case strategies exist and the best known algorithm is in $\text{NP} \cap \text{coNP}$ [14, 27, 5]. For shortest path games, where we consider game graphs with strictly positive weights and try to minimize the cost to target, it can be shown that memoryless strategies also suffice, and the problem is in P. In MDPs, optimal expectation strategies

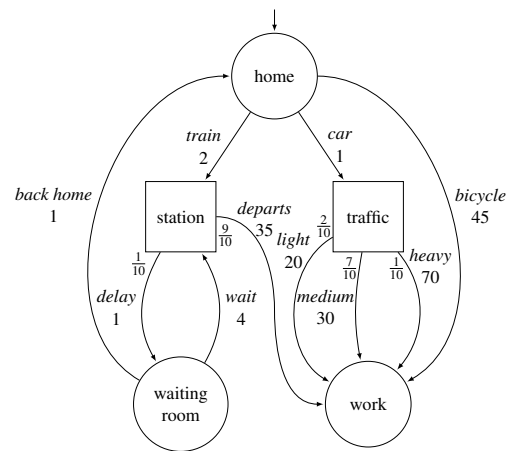


Figure 1: Player 1 wants to minimize its expected time to reach “work”, but while ensuring it is less than an hour in all cases.

are studied in [23, 15] for both measures: memoryless strategies suffice and they can be computed in P. Our strategies are *strongly risk averse*: they avoid at all cost outcomes below a given threshold (no matter their probability), and inside the set of those *safe* strategies, we maximize expectation. To the best of our knowledge, we are the first to consider such strategies. Other notions of risk have been studied for MDPs: e.g., in [26], the authors want to find policies minimizing the probability (risk) that the total discounted rewards do not exceed a specified value; in [16], the authors want to achieve a specified value of the long-run limiting average reward at a given probability level (percentile). While those strategies limit risk, they only ensure *low probability* for bad behaviors but not their absence, furthermore, they do not ensure good expectation either. Another body of related work is the study of strategies in MDPs that achieve a trade-off between the expectation and the variance over the outcomes (e.g., [4] for the mean-payoff, [22] for the cumulative reward), giving a statistical measure of the stability of the performance. In our setting, we strengthen this requirement by asking for *strict guarantees on individual outcomes*, while maintaining an appropriate expected payoff.

Future work. Study of other value functions, extension to more general settings (decidable classes of imperfect information games [13], multi-dimension [9, 11], etc), and application to practical cases.

2 Beyond Worst-Case Synthesis

Preliminaries. We consider the classical models of *games* and *MDPs*. Both are based on underlying directed *graphs* with integer weights on edges.

In *games*, the set of vertices, called states, is partitioned between states of the first player, denoted by \mathcal{P}_1 , and states of its adversary, denoted by \mathcal{P}_2 . When the game is in a state belonging to \mathcal{P}_i , $i \in \{1, 2\}$, then \mathcal{P}_i chooses a successor state according to his *strategy*, which may in general use memory (i.e., depend on the history) and be randomized (i.e., prescribe a probability distribution over successor states). This process gives rise to a *play*, an infinite sequence of states corresponding to a path through the game graph. We assign real values to plays according to a *value function*.

In *MDPs*, the set of states is partitioned between states of \mathcal{P}_1 and stochastic states, where the successor state is chosen according to a given probability distribution. Basically, an MDP is a game where the strategy of \mathcal{P}_2 is fixed.

When we fix the strategy of \mathcal{P}_1 in an MDP, or the strategies of \mathcal{P}_1 and \mathcal{P}_2 in a game, we obtain a *Markov chain* (MC), a graph where all successor states are chosen according to a stochastic transition function. Given an MC, it is well-known that measurable sets of plays have uniquely defined probabilities [25], and if we have a measurable value function, we can also compute the *expected value* or *expectation* of this function when executing the MC from a given initial state.

Classical problems. In games, the *worst-case threshold problem* asks if \mathcal{P}_1 has a strategy such that any possible outcome, against any possible strategy of \mathcal{P}_2 , gives a play with a value higher than a given threshold. In MDPs, the *expected value threshold problem* asks if \mathcal{P}_1 has a strategy such that the resulting MC yields an expectation higher than a given threshold.

Our model. The *beyond worst-case (BWC) problem* asks if \mathcal{P}_1 has a finite-memory strategy ensuring, *simultaneously*, a value greater than a threshold μ in the worst-case (i.e., against any strategy of the adversary), and an expected value greater than a threshold v against a given finite-memory stochastic model of the adversary (e.g., representing commonly observed behavior of the environment). The *BWC synthesis problem* asks to synthesize such a strategy if one exists.

3 Mean-Payoff

What was known. Given a play, its mean-payoff is defined as the (inf or sup) limit of the mean encountered weights along its finite prefixes: essentially, it is the long-run average weight over the infinite play. For the worst-case threshold problem, pure memoryless optimal strategies exist for both players [21, 14] and deciding the winner is in $\text{NP} \cap \text{coNP}$ [27, 20, 18]. Whether the problem is in P is a long-standing open problem [5, 10]. Optimal expected values in MDPs can be achieved by memoryless strategies, and the corresponding decision problem can be solved in polynomial time through linear programming [15].

Our results. We prove that surprisingly, the BWC problem matches the decision complexity of the simpler worst-case problem, even collapsing to P if the latter were proved to be in P. Hence, we enrich the modeling and reasoning power over strategies without negative impact on the complexity class.

Theorem 1. *The beyond worst-case problem for the mean-payoff value function is in $\text{NP} \cap \text{coNP}$ and at least as hard as mean-payoff games.*

Furthermore, we establish that in contrast to the worst-case and expectation problems, some memory is now needed to win in general. Nevertheless, we show that elegantly implementable strategies suffice, constructed using clever alternation between memoryless strategies based on intuitive counters.

Theorem 2. *Memory of pseudo-polynomial size may be necessary and is always sufficient to satisfy the BWC problem for the mean-payoff: polynomial in the size of the game and the stochastic model, and polynomial in the weight and threshold values.*

Some key ideas. Our solving algorithm is too complex to be presented fully in this work. Nonetheless, we here give a few hints of its cornerstones, highlighting crucial aspects of the problem.

End-components. An important part of the algorithm relies on the analysis of *end-components* (ECs) in the MDP, i.e., strongly connected subgraphs in which \mathcal{P}_1 can ensure to stay when playing against the stochastic adversary. This is motivated by two facts. First, under any arbitrary strategy, the set of states that are seen infinitely often along an outcome corresponds with probability one to an EC [12, 1]. Second, the mean-payoff function is prefix-independent, therefore the value of any outcome only depends on the states that are seen infinitely often. Hence, the expected mean-payoff that \mathcal{P}_1 can achieve on the MDP depends *uniquely* on the value obtained in the ECs. Inside an EC, we can compute the maximal expected value that can be achieved by \mathcal{P}_1 , and this value is the same in all states of the EC [15].

Classification of ECs. To be efficient w.r.t. the expected value criterion, an acceptable strategy has to favor reaching ECs with a sufficient expectation, but under the constraint that it also guarantees satisfaction of the worst-case requirement: some ECs with high expected values may still need to be avoided because they do not permit to ensure this constraint. We establish a classification of ECs based on that observation, partitioning them between *winning ECs* (WECs) and *losing ECs* (LECs). Since the total number of ECs may be exponential, providing a representative subclass of polynomial size and computing it efficiently is a crucial point to maintain the overall $\text{NP} \cap \text{coNP}$ membership.

Within a WEC. We give a particularly interesting family of strategies for \mathcal{P}_1 that both guarantee safe outcomes for the worst-case, and prove to be efficient w.r.t. the expected value. Actually, we establish that the worst-case can be guaranteed *almost for free* in the sense that we can achieve expectations arbitrarily close (but not exactly equal) to what \mathcal{P}_1 could obtain without considering the worst-case requirement at all (i.e., in a classical MDP).

To obtain this result we use a finite-memory *combined strategy*. For two well-chosen parameters $K, L \in \mathbb{N}$, it is informally defined as follows: in phase (a), play a memoryless expected value optimal

strategy for K steps and memorize $\text{Sum} \in \mathbb{Z}$, the sum of weights along these steps; in phase (b), if $\text{Sum} > 0$, go to (a), otherwise play a memoryless worst-case optimal strategy for L steps, then go to (a). In phases (a), \mathcal{P}_1 tries to increase its expectation and approach its optimal one, while in phase (b), he compensates, if needed, losses that occurred in phase (a).

The crux of the proof is to establish that adequate values of the parameters K and L exist. Essentially, K needs to be big enough so that the overall expectation is close to the optimal, but then L also needs to grow to be able to compensate sufficiently for the worst-case, hence lowering to some extent the overall expectation. Using results related to Chernoff bounds and Hoeffding's inequality in MCs [24, 19], we are able to show that the probability of having to compensate decreases exponentially when K increases, while L only needs to be polynomial in K . Overall, this implies the desired result that the parameters can be taken large enough for the strategy to be ε -optimal w.r.t. the expectation while worst-case safe.

4 Shortest Path

What was known. In this context, we consider game graphs where all weights are strictly positive, and a target set of states that \mathcal{P}_1 wants to reach while giving an upper bound on the cost to reach it. Hence the inequalities of the BWC problem are reversed. Given a play, the value function for the shortest path computes the sum of weights up to the first encounter of a state belonging to the target set, or assigning infinity if the play never reaches such a state. The worst-case threshold problem takes polynomial time, as a winning strategy of \mathcal{P}_1 should avoid all cycles (because they yield strictly positive costs), hence usage of attractors and comparison of the worst possible sum of costs with the threshold suffices. For the expected value threshold problem, memoryless strategies suffice and the problem is in P [3, 2].

Our results. In contrast to the mean-payoff case where we could maintain the complexity of the worst-case problem, we here provide an algorithm which operates in pseudo-polynomial time instead of truly-polynomial time. Nevertheless, we prove that the problem is actually NP-hard (reduction from the K^{th} largest subset problem [17]), hence establishing that a truly-polynomial time algorithm is highly unlikely.

Theorem 3. *The beyond worst-case problem for the shortest path can be solved in pseudo-polynomial time: polynomial in the size of the underlying game graph, the stochastic model of the adversary and the encoding of the expected value threshold, and polynomial in the value of the worst-case threshold. The beyond worst-case problem for the shortest path is NP-hard.*

Once again, we show that pseudo-polynomial memory is both necessary and sufficient. Recall that the example of Fig. 1 already required memory to achieve some thresholds pair for the BWC problem.

Theorem 4. *Memory of pseudo-polynomial size may be necessary and is always sufficient to satisfy the BWC problem for the shortest path: polynomial in the size of the game and the stochastic model, and polynomial in the worst-case threshold value.*

Some key ideas. The shortest path setting has a useful property: the set of all winning strategies of \mathcal{P}_1 for the worst-case threshold problem can be represented through a finite game. Indeed, we construct, from the original game G and the worst-case threshold μ , a new game G_μ such that there is a bijection between the strategies of \mathcal{P}_1 in G_μ and the strategies of \mathcal{P}_1 in the original game G that are winning for the worst-case requirement: we unfold the original graph, tracking the current value of the sum of weights up to the threshold μ , and integrating this value in the states of an expanded graph. In the corresponding game G' , we compute the set of states R from which \mathcal{P}_1 can reach the target set with cost lower than μ and we define the subgame $G_\mu = G' \upharpoonright R$ such that any path in G_μ satisfies the worst-case requirement.

Assuming that G_μ is not empty, we can now combine it with the stochastic model of the adversary to construct an MDP in which we search for a \mathcal{P}_1 strategy that ensures reachability of the target set with an expected cost lower than the expectation threshold. If it exists, it is guaranteed that it will also satisfy the worst-case requirement against any strategy of \mathcal{P}_2 thanks to the bijection evoked earlier.

Hence, in the case of the shortest path, our approach is sequential, first solving the worst-case, then optimizing the expected value among the worst-case winning strategies. Observe that such an approach is not applicable to the mean-payoff, as in that case there exists no obvious finite representation of the worst-case winning strategies.

References

- [1] L. de Alfaro (1997): *Formal verification of probabilistic systems*. Ph.D. thesis, Stanford University.
- [2] L. de Alfaro (1999): *Computing Minimum and Maximum Reachability Times in Probabilistic Systems*. In: *Proc. of CONCUR*, LNCS 1664, Springer, pp. 66–81, doi:10.1007/3-540-48320-9_7.
- [3] D.P. Bertsekas & J.N. Tsitsiklis (1991): *An analysis of stochastic shortest path problems*. *Mathematics of Operations Research* 16, pp. 580–595, doi:10.1287/moor.16.3.580.
- [4] T. Brázdil, K. Chatterjee, V. Forejt & A. Kucera (2013): *Trading Performance for Stability in Markov Decision Processes*. In: *Proc. of LICS*, IEEE Computer Society, pp. 331–340, doi:10.1109/LICS.2013.39.
- [5] L. Brim, J. Chaloupka, L. Doyen, R. Gentilini & J.-F. Raskin (2011): *Faster algorithms for mean-payoff games*. *Formal Methods in System Design* 38(2), pp. 97–118, doi:10.1007/s10703-010-0105-x.
- [6] V. Bruyère, E. Filiot, M. Randour & J.-F. Raskin (2013): *Meet your expectations with guarantees: beyond worst-case synthesis in quantitative games*. CoRR abs/1309.5439. Available at <http://arxiv.org/abs/1309.5439>.
- [7] V. Bruyère, E. Filiot, M. Randour & J.-F. Raskin (2014): *Meet your expectations with guarantees: beyond worst-case synthesis in quantitative games*. In: *Proc. of STACS*, LIPIcs, Schloss Dagstuhl - LZI.
- [8] K. Chatterjee & L. Doyen (2011): *Games and Markov Decision Processes with Mean-payoff Parity and Energy Parity Objectives*. In: *Proc. of MEMICS*, LNCS, Springer, doi:10.1007/978-3-642-25929-6_3.
- [9] K. Chatterjee, L. Doyen, T.A. Henzinger & J.-F. Raskin (2010): *Generalized Mean-payoff and Energy Games*. In: *Proc. of FSTTCS*, LIPIcs 8, Schloss Dagstuhl - LZI, pp. 505–516, doi:10.4230/LIPIcs.FSTTCS.2010.505.
- [10] K. Chatterjee, L. Doyen, M. Randour & J.-F. Raskin (2013): *Looking at Mean-Payoff and Total-Payoff through Windows*. In: *Proc. of ATVA*, LNCS 8172, Springer, pp. 118–132, doi:10.1007/978-3-319-02444-8_10.
- [11] K. Chatterjee, M. Randour & J.-F. Raskin (2012): *Strategy Synthesis for Multi-Dimensional Quantitative Objectives*. In: *Proc. of CONCUR*, LNCS 7454, Springer, pp. 115–131, doi:10.1007/978-3-642-32940-1_10.
- [12] C. Courcoubetis & M. Yannakakis (1995): *The Complexity of Probabilistic Verification*. *J. ACM* 42(4), pp. 857–907, doi:10.1145/210332.210339.
- [13] A. Degorre, L. Doyen, R. Gentilini, J.-F. Raskin & S. Torunczyk (2010): *Energy and Mean-Payoff Games with Imperfect Information*. In: *Proc. of CSL*, LNCS 6247, Springer, pp. 260–274, doi:10.1007/978-3-642-15205-4_22.
- [14] A. Ehrenfeucht & J. Mycielski (1979): *Positional Strategies for Mean Payoff Games*. *Int. Journal of Game Theory* 8(2), pp. 109–113, doi:10.1007/BF01768705.
- [15] J. Filar & K. Vrieze (1997): *Competitive Markov decision processes*. Springer, doi:10.1007/978-1-4612-4054-9.
- [16] J.A. Filar, D. Krass & K.W. Ross (1995): *Percentile Performance Criteria For Limiting Average Markov Decision Processes*. *Transactions on Automatic Control*, pp. 2–10, doi:10.1109/9.362904.

- [17] M.R. Garey & D.S. Johnson (1979): *Computers and intractability: a guide to the Theory of NP-Completeness*. Freeman New York.
- [18] T. Gawlitza & H. Seidl (2009): *Games through Nested Fixpoints*. In: *Proc. of CAV*, LNCS 5643, Springer, pp. 291–305, doi:10.1007/978-3-642-02658-4_24.
- [19] P.W. Glynn & D. Ormoneit (2002): *Hoeffding’s inequality for uniformly ergodic Markov chains*. *Statistics & Probability Letters* 56(2), pp. 143–146.
- [20] M. Jurdziński (1998): *Deciding the Winner in Parity Games is in $UP \cap co-UP$* . *Inf. Process. Lett.* 68(3), pp. 119–124, doi:10.1016/S0020-0190(98)00150-1.
- [21] T.M. Liggett & S.A. Lippman (1969): *Stochastic games with perfect information and time average payoff*. *Siam Review* 11(4), pp. 604–607, doi:10.1137/1011093.
- [22] S. Mannor & J.N. Tsitsiklis (2011): *Mean-Variance Optimization in Markov Decision Processes*. In: *Proc. of ICML*, Omnipress, pp. 177–184.
- [23] M.L. Puterman (1994): *Markov decision processes: discrete stochastic dynamic programming*, 1st edition. John Wiley & Sons, Inc., New York, NY, USA.
- [24] M. Tracol (2009): *Fast convergence to state-action frequency polytopes for MDPs*. *Oper. Res. Lett.* 37(2), pp. 123–126, doi:10.1016/j.orl.2008.12.003.
- [25] M.Y. Vardi (1985): *Automatic Verification of Probabilistic Concurrent Finite-State Programs*. In: *Proc. of FOCS*, IEEE Computer Society, pp. 327–338, doi:10.1109/SFCS.1985.12.
- [26] C. Wu & Y. Lin (1999): *Minimizing Risk Models in Markov Decision Processes with Policies Depending on Target Values*. *Journal of Mathematical Analysis and Applications* 231(1), pp. 47–67, doi:10.1006/jmaa.1998.6203.
- [27] U. Zwick & M. Paterson (1996): *The complexity of mean payoff games on graphs*. *Theoretical Computer Science* 158, pp. 343–359, doi:10.1016/0304-3975(95)00188-3.