Long-run Satisfaction of Path Properties
Christel Baier, Nathalie Bertrand, Jakob Piribauer, Ocan Sankur

To cite this version:

HAL Id: hal-02349456
https://hal.archives-ouvertes.fr/hal-02349456
Submitted on 20 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Long-run Satisfaction of Path Properties
(extended version)

Christel Baier¹, Nathalie Bertrand², Jakob Piribauer¹, Ocan Sankur²
¹Technische Universität Dresden, Germany
²Univ Rennes, Inria, CNRS, IRISA, France

Abstract—The paper introduces the concepts of long-run frequency of path properties for paths in Kripke structures, and their generalization to long-run probabilities for schedulers in Markov decision processes. We then study the natural optimization problem of computing the optimal values of these measures, when ranging over all paths or all schedulers, and the corresponding decision problem when given a threshold. The main results are as follows. For (repeated) reachability and other simple properties, optimal long-run probabilities and corresponding optimal memoryless schedulers are computable in polynomial time. When it comes to constrained reachability properties, memoryless schedulers are no longer sufficient, even in the non-probabilistic setting. Nevertheless, optimal long-run probabilities for constrained reachability are computable in pseudo-polynomial time in the probabilistic setting and in polynomial time for Kripke structures. Finally for co-safety properties expressed by NFA, we give an exponential-time algorithm to compute the optimal long-run frequency, and prove the PSPACE-completeness of the threshold problem.

I. INTRODUCTION

While the standard semantics of temporal logics relies on Boolean truth values for formulas over system models, several approaches have been studied to quantify how well a system model satisfies a temporal formula. This includes work on the robust satisfaction of temporal specifications [31], [36], vacuity and coverage semantics [15], [16], [30], [32], robustness distances [10] and the more general model-measurement semantics based on automatic distance functions of [26]. Another direction attempts to measure the degree to which a specification is satisfied when evolving over time. This includes, e.g., the work on frequency LTL [7] where a quantitative variant $a \cup_q b$ of the until operator relaxes the standard meaning of $a \cup b$ by requiring that $a$ holds at a fraction $q$ or more of the positions before $b$ holds. Other variants of frequency LTL [23], [24] allow only a quantitative variant $\Box_q$ of the globally operator. The semantics here is that $\Box_q \phi$ holds on a path if the long-run average of the frequency of positions at which $\phi$ holds is at least $q$. Alternatively, averaging LTL [8] rather than truth values, assigns quantities to pairs of paths and formula. It is based on a quantitative labeling function for atomic propositions and inductively defines the semantics of $\Box \varphi$ as the average of the value of $\varphi$ along the path. A notable similarity of these two quantitative extensions of LTL is the undecidability of the model checking problem of the full logics [7], [8]. Decidable fragments of frequency LTL can be obtained by restricting the nesting of temporal operators or the allowed frequency thresholds [7], [23], [24].

Following the spirit of quantifying the validity of a property along a path, we introduce the notion of long-run frequencies for $\omega$-regular properties. Phrased in averaging LTL words, no nesting of the averaging operators is allowed, and the labeling function is Boolean. As the name suggests, long-run frequencies measure in the long-run how frequently a property holds. For finite-state Kripke structures (KS), we study the optimization over all paths of the long-run frequency of a given property.

Fig. 1 gives two examples of KS on which one wants to evaluate the long-run frequency of an until property. Here $a, b, c$ stand for atomic propositions. For the KS on the left, the long run frequency of $a \cup b$ along e.g. the path $(abc)^\omega$ is $\frac{2}{3}$. The maximal long-run frequency is 1, which is achieved, e.g., by the infinite path $abca^2bca^3bca^4bc \ldots$ that successively doubles the number of times the self-loop at state $a$ is taken. However, there is no finite-memory strategy for generating an infinite path where the long-run frequency for $a \cup b$ is 1. The KS on the right illustrates, that, even when infinite-memory is not needed, memoryless is not
enough: for $\alpha \mathcal{U} b$, the maximal long-run frequency is achieved by alternating between the two simple cycles and amounts to $\frac{3}{2}$, which is indeed more than $\frac{2}{3}$, the long-run frequency of iterating the bottom cycle only.

When turning to the probabilistic world, we introduce the corresponding concept of long-run probabilities. On Markov chains, long-run probabilities are limit-average probabilities for path properties, indicating the probability for a property to hold on the suffix of a path after many steps. They can, among others, serve to provide refined measures for the system availability, understood as the proportion of time a system is functioning under “normal” operating conditions (after the initialization phase). For finite Markov decision processes (MDP), the corresponding optimization problem is to compute the optimal long-run probability of a given property, when ranging over all schedulers, or to decide how this value compares to a threshold.

To illustrate the notion of long-run probability, consider the MDP $N_k$ shown in Fig. 2, the only non-determinism is between actions $\alpha$ and $\beta$, and $\alpha$ yields a uniform distribution over the three successors.

![Fig. 2. MDP with labels indicated by the state names requiring counting to maximize the long-run probability of $\alpha \mathcal{U} b$.](image)

Under the memoryless scheduler $\Sigma_\alpha$ that always picks action $\alpha$, the probability of $\alpha \mathcal{U} b$ in the $a$-state is $\frac{1}{2}$, and its frequency is $\frac{3}{2}$. The state $b_1$ has frequency $\frac{1}{2}$ and from there the probability of $\alpha \mathcal{U} b$ is 1. We thus compute the long-run probability under $\Sigma_\alpha$ to be $\frac{1}{2}$. Similarly, the steady-state probability of the states $a$ and $b_2$ under the memoryless scheduler $\Sigma_\beta$ are $\frac{1}{k+2}$, and the probability that $\alpha \mathcal{U} b$ holds from there is 1. The long-run probability of $\alpha \mathcal{U} b$ under $\Sigma_\beta$ equals $\frac{2}{k+2}$. Observe that the satisfaction probability at the $a$-state depends on the scheduler (it is $\frac{1}{2}$ for $\Sigma_\alpha$ and 1 for $\Sigma_\beta$). The intricate interaction of satisfaction probability and frequency of each state makes the optimization of long-run probability particularly challenging. Here, we will see that counting the number of consecutive visits to the $a$-state allows one to derive a scheduler that achieves a higher long-run probability than the two memoryless ones.

**Contributions:** Beyond the introduction of the notions of long-run frequency and long-run probability, in this paper, we establish complexity bounds for the computation of the value (optimal long-run frequency or probability) and the associated threshold problem. These are summarized in Table I, split depending on the type of properties and the models (KS or MDP). In particular, computing the optimal long-run probability for simple properties (such as reachability, invariant, Rabin or Streett conditions) can be done in polynomial time for MDP, in which case optimal memoryless schedulers exist. This entails the same complexity upper bound for the particular case of computing long-run frequency of these simple properties in KS. Moreover, the computation of the optimal long-run frequency in KS for constrained reachability properties (expressed by until formulas) can also be done in polynomial time, although, as explained already, infinite memory can be necessary. Our main contribution for non-probabilistic structures, is an exponential time algorithm for computing optimal long-run frequency for regular co-safety properties (specified by an NFA). It is obtained by reducing to the computation of the optimal mean-payoff in an exponentially large weighted KS. We also prove the PSPACE-completeness of the corresponding threshold problem.

In comparison, the probabilistic setting is substantially harder, already for constrained reachability properties, expressed by until formulas. As in our illustrating example, and contrary to the case of the simple properties mentioned above, maximizing in each state the probability that an until property holds does not yield the maximal long-run probability. Also when finite-memory schedulers are optimal, as opposed to the non-probabilistic case where memory with two modes suffices, a counter up to some bound that depends on the size of the description of the MDP is needed. Proving the existence of this saturation point (the ideal number of consecutive visits to $a$-states for the property $\alpha \mathcal{U} b$) is the crux to derive our pseudo-polynomial time algorithm for computing maximal or minimal long-run probabilities. The corresponding threshold problem is shown to be NP-hard. These two results certainly constitute the most involved contribution of the paper. We also show that the corresponding questions for qualitative threshold problems (e.g., whether the maximal long-run probability for an until property is positive, or is 1) are solvable in polynomial time.

**Related work:** We mentioned the quantitative semantics of LTL and the decidable fragments of frequency LTL with a quantitative globally operator $\square_q$ [23], [24] which are the closest to our work. Frequency LTL, however, is a logic to specify quantitative measures for the satisfaction of properties along paths using the $\square_q$-modality, while long-run probabilities are a quantitative measure across behaviors. For finite strongly connected Markov chains, the probabilities for $\square_q$-formulas are 0 or 1, while long-run probabilities can be strictly between 0 and 1. There is still a connection as for each finite, strongly connected Markov chain $M$, $\square_q(\alpha \mathcal{U} b)$ holds in $M$ with probability 1 iff the long-run probability of $\alpha \mathcal{U} b$ is at least $q$. Nevertheless,
the contribution for MDPs in [23], [24] are orthogonal to ours. On the one hand, they can treat much more complex properties with nested □ₚ-formulas. On the other hand, they cannot deal with formulas of the type □ₚ(αUβ) for p < 1. The results in [23] only apply to p = 1. The fragment in [24] can deal with □ₚ-modalities for arbitrary p, but imposes the constraint that no until operator occurs in the scope of the □ₚ-modality.

Despite many works on long-run properties in MDPs (e.g., mean payoff [9], [11], [27] and other cost objectives [20] or ratios [19], [37]), we are not aware that long-run probabilities for MDPs have been studied before. Long-run probabilities can be seen as mean-payoff, where the weights are the satisfaction probabilities. A crucial difference however with mean-payoff and other long-run properties is that, for long-run probabilities, the “weights” along a path are not fixed a priori, but do depend on the scheduler. In this aspect, there is some conceptual relation to dynamic Markov processes [34] where cost or transition probabilities depend on previously made decisions, or the stochastic variant of the Canadian traveler problem [25]. These problems, however, are concerned with finite-horizon objectives; moreover, their weights are affected by the past, whereas our “weights” (satisfaction probabilities) are induced by the future scheduler.

Outline: Section II summarizes the notations used in the paper. Our results for non-probabilistic systems are presented in Section III, while Section IV discusses long-run probabilities in MDPs. We conclude in Section V. Proofs can be found in the appendix.

II. PRELIMINARIES

We suppose familiarity with linear temporal logic (LTL), Kripke structures, finite automata, and basic concepts of discrete Markovian models, and only provide a summary of the notations used in the paper. Details can be found in textbooks, e.g., [3], [17], [35].

Non-deterministic finite automata (NFA): An NFA is a tuple $A = (Q, \Sigma, \delta, Q_0, F)$ where $Q$ is a finite set of states, $\Sigma$ an alphabet, $\delta \subseteq S \times \Sigma$ the transition relation, $Q_0 \subseteq Q$ the set of initial states and $F \subseteq Q$ the set of final states. $L(A)$ is the accepted language of $A$.

Kripke structures (KSs): A KS is a tuple $\mathcal{T} = (S, \Delta, AP, L)$ where $(S, \Delta)$ is a finite directed graph, $AP$ a finite set of atomic propositions and $L : S \rightarrow 2^{AP}$ a labeling function. The trace of a path $\pi = s_0 s_1 s_2 \ldots$ is the word $L(\pi) = L(s_0) L(s_1) L(s_2) \ldots$ over $2^{AP}$ obtained by projecting states to their labels. If $\pi = s_0 s_1 s_2 \ldots$ is a path then we write first$(\pi)$ for its first state $s_0$, $\pi_{[i]}$ for the $(i+1)$-st state $s_i$, and $\pi_{[i..j]}$ for the path fragment $s_i s_{i+1} \ldots s_j$. Likewise, $\pi_{[i..]}$ and $\pi_{[j..]}$ stand for the prefix ending in state $s_i$ resp. the suffix starting from $s_j$. If $T \subseteq S$, a $T$-state is a state in $T$, and a $T$-cycle is a cycle consisting of $T$-states.

We shall use LTL-like notations to denote path properties. For instance, if $T$ is a set of states then $\diamond T$ stands for the event “eventually reaching $T$”, $\square T$ for the event “always $T$” and $\diamond \square T$ stands for Büchi condition “infinitely often $T$”. The modality $U$ stands for the standard until operator. Likewise, CTL-like notations are used for state properties, e.g., $s \models \exists T$ indicates that a $T$-state is reachable from state $s$.

Weighted structures: A weighted KS $\mathcal{T}$ extends a plain KS $\mathcal{T}$ as above by a weight function $\text{wgt} : \Delta \rightarrow Q$ that assigns rational values to transitions. We also use weight functions on states (rather than transitions), which can be seen as a special case of transition-based weight functions. Given an infinite path $\pi = s_0 s_1 \ldots$ in a weighted KS $\mathcal{T}$, the mean payoff $\text{mp}(\pi)$ is defined by:

$$\text{mp}(\pi) = \liminf_{n \to \infty} \frac{1}{n+1} \sum_{i=0}^{n} \text{wgt}(s_i) .$$

The maximal mean payoff from state $s$, $\text{Mmp}^{\mathcal{T}}_{\text{max}}$, is $\sup_{\pi} \text{mp}(\pi)$ where $\pi$ ranges over all infinite paths starting at $s$. For $I \subseteq S$ we define $\text{Mmp}^{\mathcal{T}, I}_{\text{max}} = \max_{s \in I} \text{Mmp}^{\mathcal{T}}_{\text{max}}$. Analogous notations are used for the minimal mean payoff.

<table>
<thead>
<tr>
<th>reachability/invariant/ Rabin/Streett conditions</th>
<th>non-probabilistic case</th>
<th>probabilistic case</th>
</tr>
</thead>
<tbody>
<tr>
<td>value computable in polynomial-time</td>
<td>value computable in exponential time</td>
<td></td>
</tr>
<tr>
<td>(special case of Theorem IV.3)</td>
<td>(Corollary III.6)</td>
<td></td>
</tr>
<tr>
<td>constrained reachability (αUβ)</td>
<td>value computable in polynomial-time</td>
<td></td>
</tr>
<tr>
<td>value computable in polynomial-time</td>
<td>qualitative decision problems in polynomial-time</td>
<td></td>
</tr>
<tr>
<td>(Corollary III.3)</td>
<td>(Lemma IV.5)</td>
<td></td>
</tr>
<tr>
<td>regular co-safety (NFA)</td>
<td>value computable in pseudo-polynomial time</td>
<td></td>
</tr>
<tr>
<td>value computable in exponential time</td>
<td>(Theorem IV.10)</td>
<td></td>
</tr>
<tr>
<td>(Corollary III.6)</td>
<td>NP-hard threshold problem</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Theorem IV.11)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>computability of the value: open</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(corollary of Theorem III.7)</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE I**

**SUMMARY OF THE MAIN RESULTS.**
Markov decision processes (MDPs): An MDP is a tuple \( M = (S, Act, \mathcal{P}) \) where \( S \) is a finite state space, \( Act \) a finite nonempty set of actions and \( \mathcal{P} : S \times Act \times S \to [0, 1] \cap \mathbb{Q} \) the transition probability function satisfying \( \sum_{t \in S} \mathcal{P}(s, \alpha, t) = 1 \) for all \( (s, \alpha) \in S \times Act \). The triples \((s, \alpha, t)\) with \( \mathcal{P}(s, \alpha, t) > 0 \) are called transitions of \( M \). The actions enabled in \( s \) form \( Act(s) = \{ \alpha \in Act : \sum_{t \in S} \mathcal{P}(s, \alpha, t) = 1 \} \). We often refer to the pairs \((s, \alpha)\) with \( \alpha \in Act(s) \) as the state-action pairs of \( M \). MDPs can be augmented with atomic propositions labeling states and weight functions, as for KSs. The size of an MDP is the number of states and actions plus the sum of the logarithmic lengths of the transition probabilities. For a weighted MDP, we add the sum of the logarithmic lengths of the transition probabilities.

Intuitively, when \( M \) is at a state \( s \), then an action \( \alpha \) of \( Act(s) \) is selected nondeterministically; afterwards the next state is obtained by probabilistically choosing one of the potential successor states according to the probability distribution \( \mathcal{P}(s, \alpha, \cdot) \). Paths in MDP are alternating sequences of states and actions: \( \pi = s_0 \alpha_0 s_1 \alpha_1 \ldots \) where \( \alpha_i \in Act(s_i) \) and \( \mathcal{P}(s_i, \alpha_i, s_{i+1}) > 0 \) for all \( i \geq 0 \). The notations \( first(\pi) \), \( \pi_{[i..j]} \) and \( \overline{\pi} \) are defined as for paths in KS. End components (ECs) of an MDP are strongly connected sub-MDPs: they are formed of sets of state-action pairs where the induced graph is strongly connected. Maximal ECs (MECs) are ECs not contained in other ECs.

A scheduler \( \mathcal{G} \) for \( M \) is a function that maps a finite path \( \overline{\alpha} \) to a probability distribution over \( Act(last(\overline{\alpha})) \) (where \( last(\overline{\alpha}) \) is the last state of \( \overline{\alpha} \)). Given a finite path \( \overline{\alpha} \), then \( \mathcal{G} \uparrow \overline{\alpha} \) denotes the residual scheduler defined by \( (\mathcal{G} \uparrow \overline{\alpha})(\overline{\alpha}') = \mathcal{G}(\overline{\alpha}; \overline{\alpha}') \) for each finite path \( \overline{\alpha}' \) starting in \( last(\overline{\alpha}) \). \( \mathcal{G} \) is deterministic if \( \mathcal{G}(\overline{\alpha}) \) is a Dirac distribution for each finite path, in which case \( \mathcal{G} \) can be viewed as a function that maps finite paths to actions. Finite-memory schedulers are those that can be realized using a finite-state automaton whose states, called modes, serve as memory cells (see e.g. [3] for the formal definition). The decisions of memoryless schedulers only depend on the last state. They can be viewed as functions from states to distributions over actions. We write HR for the full class of (history-dependent randomized) schedulers, FMR or briefly FM (resp. FMD) for the class of finite-memory randomized (resp. deterministic) schedulers and MD for the class of memoryless deterministic schedulers.

\( \Pr_{\overline{\alpha}}^{\mathcal{G}, s} \) denotes the probability measure induced by \( \mathcal{G} \), when \( s \) is the initial state. It is well-known that all \( \omega \)-regular path properties \( \varphi \) are measurable and there exist FMD-schedulers maximizing or minimizing the probability for \( \varphi \). This justifies the notations \( \Pr_{\overline{\alpha}}^{\max} \) and \( \Pr_{\overline{\alpha}}^{\min} \) for \( \omega \)-regular properties. We consider the maximal (resp. minimal) expected mean payoff, denoted \( \mathbb{E}_{\overline{\alpha}}^{\max}(M\varphi) \) and \( \mathbb{E}_{\overline{\alpha}}^{\min}(M\varphi) \), where the extrema are taken over all schedulers. It is well-known that optimal MD-schedulers for mean payoff objectives exist and are computable in polynomial time (see e.g. [35]). States belonging to the same MEC have the same maximal (resp. minimal) probability for reachability and prefix-independent properties and the same extremal expected mean payoff. This justifies notations like \( \Pr_{\overline{\alpha}}^{\max}(\varphi)(b) \) or \( \Pr_{\overline{\alpha}}^{\min}(M\varphi) \) for ECs \( \mathcal{E} \). As shown in [19], for each scheduler \( \mathcal{G} \) and each state \( s \), the limit of almost all infinite paths constitutes an EC.

**Remark II.1 (Traps).** When studying long-run frequencies and long-run probabilities, we assume the given KSs and MDPs have no traps, i.e., states with no outgoing transitions. This ensures the existence of infinite paths from every state. The presented reductions, however, can generate structures with traps, in which case optimal strategies in the constructed structures need to avoid these (trap) states.

**III. Long-run frequencies in non-probabilistic systems**

Let \( \mathcal{T} = (S, \Delta, AP, L) \) be a KS and \( \varphi \) a path property. The long-run frequency for \( \varphi \) along an infinite path \( \pi \) of \( \mathcal{T} \) is defined as:

\[
\text{lf}_\varphi(\pi) = \liminf_{n \to \infty} \frac{1}{n+1} \sum_{i=0}^{n} \mathbb{I}_{\pi_{[i..j]}}(\varphi)
\]

where \( \mathbb{I}_{\pi_{[i..j]}}(\varphi) = 1 \) if \( \pi_{[i..j]} \models \varphi \) and 0 otherwise. The problem we address is how to compute the maximal long-run frequency for \( \varphi \) given by

\[
\text{lf}_\varphi^{\max}(\varphi) = \sup_{\pi} \text{lf}_\varphi(\pi)
\]

where \( s \in S \) and \( \pi \) ranges over all infinite paths starting in state \( s \), and the analogously defined minimal long-run frequency \( \text{lf}_\varphi^{\min}(\varphi) \).

Obviously, the value \( \text{lf}_\varphi^{\max}(\varphi) \) is prefix-independent, and all states belonging to the same strongly connected component (SCC) of \( \mathcal{T} \) have the same extremal values. It thus suffices to determine the optimal values for the SCC of \( \mathcal{T} \). The optimal value for a given state \( s \) of \( \mathcal{T} \) is then the maximum or minimum of the optimal values for the SCCs reachable from \( s \). In the sequel we therefore assume \( \mathcal{T} \) is strongly connected, and simply write \( \text{lf}_\varphi^{\max}(\varphi) \) and \( \text{lf}_\varphi^{\min}(\varphi) \).

As a consequence of a result established later for the probabilistic setting (see Theorem IV.3), the extremal long-run frequencies for invariants, reachability, Rabin and Streett conditions are computable in polynomial time. For KS, these techniques essentially require to identify “good” cycles \( \xi \), where the property under consideration holds from all states along \( \xi \).

Reasoning about long-run frequencies becomes more challenging when considering properties that are not
A. Long-run frequencies for until properties

Before presenting the polynomial-time computation scheme for extremal long-run frequencies of until properties, we introduce useful notations, and identify easy particular cases. Fix a strongly connected KS $\mathcal{T}$, and the until property $\varphi = a\mathbf{U}b$ with $a, b \in \mathcal{AP}$. Let

\[
\begin{align*}
A &= \{ s \in S \mid s \models \exists (aUb), s \notin \forall (aUb) \}, \\
B &= \{ s \in S \mid s \models \forall (aUb) \}, \\
C &= S \setminus (A \cup B).
\end{align*}
\]

If $B = \emptyset$, $\mathcal{L}_{\mathcal{T}}^{\text{max}}(aUb) = \mathcal{L}_{\mathcal{T}}^{\text{min}}(aUb) = 0$, thus we assume $B \neq \emptyset$. The until properties $aUb$ and $AUB$ are equivalent: for each infinite path $\pi$ of $\mathcal{T}$, $\pi \models aUb$ iff $\pi \models AUB$, and therefore $lrf_{aUB}(\pi) = lrf_{AUB}(\pi)$.

Using these sets, we first characterize easy cases:

**Lemma III.1.** $\mathcal{L}_{\mathcal{T}}^{\text{max}}(aUb) = 1$ iff $\mathcal{T}$ has a $A \cup B$-cycle. $\mathcal{L}_{\mathcal{T}}^{\text{min}}(aUb) = 0$ iff $\mathcal{T}$ has a $A \cup C$-cycle.

As a consequence of Lemma III.1 (see Lemma A.1), if $\mathcal{T}$ contains an $A$-cycle then the values of $\mathcal{L}_{\mathcal{T}}^{\text{max}}(aUb)$ and $\mathcal{L}_{\mathcal{T}}^{\text{min}}(aUb)$ are respectively 1 and 0. This observation permits the additional assumption that $\mathcal{T}$ has no $A$-cycles.

The computation of $\mathcal{L}_{\mathcal{T}}^{\text{max}}(aUb)$ and $\mathcal{L}_{\mathcal{T}}^{\text{min}}(aUb)$ reduces to the computation of the extremal mean-payoff in a polynomial size weighted KS. The idea of the reduction is to deal with two copies of $A$-states. Intuitively, when entering an $A$-state there is a nondeterministic choice between mode 0 and mode 1 where the former expects $AUB$ to be violated, while the latter expects some $B$-state will be visited after consecutive $A$-states. From any state $(s, 0)$ with $s \in A$, the transitions to $B$-states are removed. Likewise, from states $(s, 1)$ with $s \in A$ there are no transitions to $C$-states. Since $\mathcal{T}$ has no $A$-cycles, the sub-graph consisting of states in $A \times \{0, 1\}$ is acyclic. The weight function in $\mathcal{T}^*$ is state-based and assigns weight 1 to the states in $B \cup A \times \{1\}$ and weight 0 to the states in $C \cup A \times \{0\}$. The construction is illustrated on Fig. 3 for the KS right of Fig. 1. The soundness of the construction is stated in the following lemma (see Appendix, Lemma A.2 for its proof).

**Lemma III.2.** Suppose $\mathcal{T}$ has no $A$-cycles. Then,

$$
\mathcal{L}_{\mathcal{T}}^{\text{max}}(aUb) = \mathcal{L}_{\mathcal{T}}^{\text{max}}(A) \quad \text{and} \quad \mathcal{L}_{\mathcal{T}}^{\text{min}}(aUb) = \mathcal{L}_{\mathcal{T}}^{\text{min}}(A).
$$

Thus, the extremal long-run frequencies for until properties are computable using standard techniques for weighted KS with mean payoff objectives. This yields:

**Corollary III.3.** The values $\mathcal{L}_{\mathcal{T}, s}^{\text{max}}(A \cup B)$ and $\mathcal{L}_{\mathcal{T}, s}^{\text{min}}(aUb)$ are computable in polynomial time.

**Remark III.4 (Memory requirements).** The example from Fig. 1 left in the introduction shows that infinite memory can be necessary to optimize the long-frequency for until-properties. If, however, $\mathcal{T}$ does not contain $A$-cycles, then two memory cells suffice for optimizing the long-run frequency for $AUB$. (This follows from Lemma III.2 and the well-known fact that optimal memoryless strategies for mean payoff objectives exist.) This stands in great contrast with the probabilistic case, as we will see in Section IV.

B. Regular co-safety properties

We now address extremal long-run frequencies in KS for regular co-safety properties $\varphi$. Fix $\mathcal{T}$ a strongly connected KS, and let $A = (Q, \Sigma, \delta, Q_0, F)$ be an NFA over the alphabet $\Sigma = \mathcal{AP}$ representing $\varphi$, i.e., an infinite path of $\mathcal{T}$ satisfies $\varphi$ iff it has a prefix accepted by $A$. Hence, $lrf_{\varphi}(\pi)$, also denoted $lrf_{A}(\pi)$, is the long-run average of positions in $\pi$ where a word in $\mathcal{L}(A)$ starts, and we write $\mathcal{L}_{\mathcal{T}}(A)$ rather than $\mathcal{L}_{\mathcal{T}}(\varphi)$. We show that the computation of $\mathcal{L}_{\mathcal{T}}^{\text{max}}(A)$ and $\mathcal{L}_{\mathcal{T}}^{\text{min}}(A)$ reduces to determine the maximal and minimal mean-payoff in a weighted KS $\mathcal{S}$ with a generalized Büchi side condition, whose size is exponential in the size of $\mathcal{T}$.

For simplicity, we suppose here that $Q_0 = \{q_0\}$ is a singleton and that $q_0$ is not accessible from any other state in $A$. We also assume that $q_0 \not\in F$ (otherwise $A$ accepts the empty word and the long-run frequency of $\varphi$ along any infinite path is 1). We fix an arbitrary state $s_0 \in S$ which we treat as a starting state of $\mathcal{T}$. (Since $\mathcal{T}$ is strongly connected, the extremal long-run frequencies in $\mathcal{T}$ do not depend on the choice of the starting state.) We define a weighted KS $\mathcal{S}$ as follows.
Let $\ell = |Q|$ denote the number of states in the NFA $A$. Then, the state space $S_G$ of $G$ is equal to

$$S \times \{\emptyset\} \cup \{\text{merged}\} \times \{0, \ldots, \ell\} \cup \{1, \ldots, 2^\ell\}.$$ 

Each state of $G$ tracks a state of $T$ and in addition a vector of $2^\ell + 1$ elements, which we explain now. For a state $(s, f) \in S_G$, $f(i)$ denotes the $i$-th component and is called the $i$-th track. Intuitively, at every step of the execution, the construction guesses whether $\varphi$ holds from the current position and creates a new track to check this guess. Hence, the symbol $\emptyset$ is used for tracks that are not in use; $(q, \text{true})$ means that the track is at state $q \in A$ and that the final state of $A$ will be eventually reached (confirming the guess that $\varphi$ holds), while $(q, \text{false})$ means that no final state of $A$ should be reached at this track (expressing the guess that $\varphi$ does not hold). The symbol merged is used to merge tracks that arrive at the same states of $A$ with the same guess. The formal details follow.

The set $I_G$ of initial states in $G$ is given by:

$$I_G = \{ (s_0, f) \mid f(0) \in \{(q_0, \text{true}), (q_0, \text{false})\} \text{ and } f(i) = \emptyset \text{ for } 1 \leq i \leq 2\ell \}.$$ 

That is, a single track is created at initial state $q_0$ of $A$ that is either true or false. The non-determinism in the choice of the initial state allows the execution to guess whether $\varphi$ holds at this position. The definition of transitions also uses non-determinism for two purposes: (i) to guess whether $\varphi$ holds at the given step and (ii) to guess a successor state in $A$. Note that for each track of the form $(q, \text{true})$, the construction will guess some successor $q'$ of $q$ in $A$ in an attempt to prove the guess correct, while for each track of the form $(q, \text{false})$, it will check all successors of $q$ in order to prove that the word that is being read is not accepted by $A$.

We define the transitions in two steps. The first step consists in allowing the tracks to make progress non-deterministically, while in the second step we merge tracks at identical states and add the new track $q_0$ for the current position.

For any state $(s, f)$ in $G$ and each state $s'$ of $T$, we define relation $\rightarrow_1$ as follows: we have $(s, f) \rightarrow_1 (s', f', H)$ where $H = \bigcup_{q \in \Delta} 3 [\xi \vdash L(f(i) - (q, \text{false})) \in \delta(q, L(s))]$, if, and only if, the following holds for all $0 \leq i \leq 2\ell$:

- If $f(i) \in \emptyset, \{\text{merged}, j\}$ for some $j$, then $f'(i) = \emptyset$.
- If $f(i) = (q, b)$ for some $(q, b) \in (Q \setminus F) \times \{\text{true}, \text{false}\}$, then $f'(i) = (q', b')$ where $q' \in \delta(q, L(s))$.
- If $f(i) = (q, \text{true})$ for some $q \in F$, then $f'(i) = \emptyset$.
- If $f(i) = (q, \text{false})$ for some $q \in F$, then $f'(i) = (q, \text{false})$.

Here, the additional component $H$ is the set of all successors in $A$ of states $q$ which appear in $f$ as $(q, \text{false})$. In fact, these must be checked for non-acceptance, and in the next step, new tracks $(q, \text{false})$ will be created for each $q \in H$ unless they already exist or they contradict another guess (i.e., if some track with $(q, \text{true})$ exists).

The intermediary state $(s', f', H)$ is called inconsistent if there exists $0 \leq i, j \leq 2\ell$ and $q \in Q$ such that one of the conditions hold:

1. $f'(i) = (q, \text{false})$ with $q \in F$.
2. $f'(i) = (q, \text{true})$ and $f'(j) = (q, \text{false})$.
3. $f'(i) = (q, \text{true})$ for some $q \in H$.

Assume $(s', f', H)$ is consistent. Let us write $H' = \{q \in H \mid \forall i. f'(i) \neq (q, \text{false})\} = \{q_1, \ldots, q_k\}$.

We define the relation $\rightarrow_2$ by $(s', t', H) \rightarrow_2 (s', t'')$ if, and only if, the following conditions hold for all $0 \leq i \leq 2\ell$:

- Case $f'(i) = \emptyset$: if $i$ is the first such position then $t''(i) = (q_0, b)$ for some $b \in \{\text{true}, \text{false}\}$, if it is the $1 + 1$-th such position, we let $t''(i) = q_1$, and otherwise $t''(i) = \emptyset$.
- Case $f'(i) = (q, b)$ for some $(q, b) \in Q \times \{\text{true}, \text{false}\}$: if for all $0 \leq j < i$, $f'(j) \neq (q, b)$, then $t''(i) = (q, b)$; and otherwise, $t''(i) = (\text{merged}, j)$ where $j$ is the least index of a track containing $(q, b)$.

Hence, the second step is used to stop all redundant tracks; for each pair $(q, b)$ only the first track survives, and others are stopped and labeled by $(\text{merged}, j)$ with $j$ being the position of the surviving track. Making the positions $j$ visible in the construction is not necessary but will be useful for the proofs. In addition, all states $q \in H$ that do not already appear in some track are added as a new track $(q, \text{false})$, and a new track is started from $q_0$.

The transitions of $G$ are defined as follows. We have $((s, f), (s', f'')) \in \Delta_G$ if and only if $(s, s') \in \Delta$ and there exists a consistent state $(s', f', H)$ such that $(s, f) \rightarrow_1 (s', f', H) \rightarrow_2 (s'', f'')$.

The weight function of $G$ is state-based and given by $\text{wgt}(s, f) = 1$ if there exists some $i$ with $f(i) = (q_0, \text{true})$, and $\text{wgt}(s, f) = 0$ otherwise.

For each $0 \leq i \leq 2\ell$, let us define the following labels. Let $F_1, \text{false}_1, \text{merged}_1,$ and $\emptyset_1$ denote the states $(s, f)$ satisfying, respectively, $f(i) = (q, \text{true})$ for some $q \in F$, $f(i) = (q, \text{false})$ for some $q \in Q$, $f(i) = (\text{merged}, j)$ for some $0 \leq j < i$, or $f(i) = \emptyset$.

We now show that the extremal long-run frequencies for co-safety properties are computable using the above construction in combination with techniques for one-player mean payoff games with generalized Büchi conditions. For this, we define $\Phi$ denote the following generalized Büchi condition:

$$\Phi = \bigwedge_{i=0}^{2\ell} \Diamond (\text{false}_1 \lor \text{merged}_1 \lor F_1 \lor \emptyset_1).$$

Let $\text{MP}_{\max}^G(\Phi)$ denote the maximal mean payoff in $G$ starting in one of the two initial states in $I_G$ while sat-
isifying $\Phi$. Similarly, $\mathcal{M}^{\min}(\Phi)$ denotes the minimal payoff in $\mathcal{G}$ while satisfying $\Phi$. The soundness of our construction is stated as follows.

**Theorem III.5.** For $\mathcal{G}$ and $\Phi$ defined as above, \(\mathcal{L}^{\max}_T(A) = \mathcal{M}^{\max}(\mathcal{G})\) and \(\mathcal{L}^{\min}_T(A) = \mathcal{M}^{\min}(\mathcal{G})\).

We already explained that the tracks in $\mathcal{G}$ guess and check the satisfaction or violation of $\varphi$ from each position. The additional Büchi condition is used to make sure that these checks are conclusive: each track of the form $(q, \text{true})$ must eventually either end in an accepting state $(F_1)$ or merge with a track with strictly lower index (which will itself reach an accepting state or merge with even smaller indexed-track). Other tracks may stay in $(\cdot, \text{false})$ or not be used $(\emptyset_1)$. A detailed proof of the above theorem is given in Theorem A.4 in the Appendix.

Using [6], [28], one can compute extremal mean payoff values in one-player games in polynomial time in the size of the game. We apply this to $\mathcal{G}$ which yields the following result (See also Appendix B):

**Corollary III.6.** \(\mathcal{L}^{\max}_{\mathcal{T}, s}(\varphi)\) and \(\mathcal{L}^{\min}_{\mathcal{T}, s}(\varphi)\) are computable in time exponential in the size of $A$ and polynomial in the size of $T$.

We now establish the complexity of the decision problem associated with the maximization of long-run frequency for co-safety properties. Formally, given a KS $T$, an NFA $A$ and a rational threshold $\vartheta$, the threshold problem asks whether $T$ admits an infinite path $\pi$ with $l_{r_{T}}(\pi) \geq \vartheta$.

**Theorem III.7.** The threshold problem for co-safety properties in KS is PSPACE-complete.

**Proof Sketch.** The PSPACE upper bound is obtained by providing a nondeterministic guess-&-check algorithm that guesses a reachable state $s_0$ in $\mathcal{G}$ and two cycles $\xi_{\Phi}$ and $\xi_{MP}$ containing $s_0$ and satisfying certain length restrictions and $\xi_{\Phi}^{m} = \Phi$ and $mp(\xi_{MP}^{m}) \geq \vartheta$. In fact, one can construct the desired path by alternating between $\xi_{\Phi}$ and $\xi_{MP}$ by making sure the former appears infinitely often but with frequency 0.

The PSPACE lower bound follows by a polynomial reduction from the intersection problem for deterministic finite automata (DFA); given k DFA $\mathcal{D}_1, \ldots, \mathcal{D}_k$ over the same alphabet $\Sigma$, is the intersection language $\mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$ nonempty? This problem is known to be PSPACE-complete [29]. A detailed proof is given in Theorem A.5 in the Appendix.

**Proposition III.8 (Qualitative thresholds).**

Deciding the existence of an infinite path $\pi$ in $T$ with

- $l_{r_{T}}(\pi) > 0$ is in PTIME;
- $l_{r_{T}}(\pi) = 0$ (resp. = 1) is NP-hard;
- $l_{r_{T}}(\pi) < 1$ is PSPACE-hard.

**Proof sketch.** The existence of an infinite path $\pi$ in $T$ with $l_{r_{T}}(\pi) > 0$ is equivalent to the existence of a finite path in (an SCC of) $T$ that is accepted by $A$. The latter can be checked by performing a graph analysis of the product $T \otimes A$.

To prove the two NP-hardness results, we provide a polynomial reduction from 3SAT. The construction of the KS $T$ is the same in both reductions: it runs through each clause one after the other and selects a literal. The constructed automata slightly differ in the two cases. For example, for the existence of a path with long-run frequency 0, the automaton accepts behavior where the choice of literals to be true is conflicting.

For the last hardness result, we reduce the universality of NFA to the problem whether for all infinite paths $\pi$ in a KS $l_{r_{A}}(\pi) = 1$, and use the fact that PSPACE is closed under complementation. Given an NFA $\mathcal{B}$ over the alphabet $\Sigma$, we construct an NFA $A$ over the alphabet $\Sigma \cup \{\#\}$ for a new symbol # such that $A$ accepts $\Sigma \cup \{\#w\#|w \in L(\mathcal{B})\}$. The Kripke structure that generates all possible sequences of letters over $\Sigma \cup \{\#\}$ then has a path $\pi$ with $l_{r_{A}}(\pi) < 1$ iff $\mathcal{B}$ is non-universal. See Lemma A.8 for the full proofs.

**IV. Long-run Probabilities in MDP**

When turning to probabilistic models, long-run probabilities generalize long-run frequencies, and the objective is to compute the optimal values of these long-run averages, when ranging over all schedulers. Our first contribution is the identification of efficiently solvable instances, including prefix-independent properties (such as Rabin or Streett conditions) where the satisfaction only depends on the states that are visited infinitely often and that can be treated by a polynomial-time analysis of end components. Our second contribution, and main result of this section is a pseudo-polynomial algorithm for computing extremal long-run probabilities for constrained reachability (until properties $a \cup b$). This result can be seen as a first step towards the treatment of more general co-safety properties in MDP and serves to illustrate which major extra difficulties arise when switching from long-run frequencies in KS to long-run probabilities in MDP.

It is important to emphasize that the computation of optimal long-run probabilities for constrained reachability does not easily reduce to reachability via a preprocessing of the MDP, as it typically does for most verification problems. Also, the traditional reduction to the case of a Rabin condition for the treatment of arbitrary $\omega$-regular properties fails here. These highlight the challenge and specificity in computing long-run probabilities.
Throughout this section, we suppose that we are given an MDP $M = (S, Act, P, AP, L)$ and a path property $\varphi$. The long-run probability for $\varphi$ of an infinite path $\pi$ under a scheduler $\mathcal{S}$ for $M$ is defined as the average of the probabilities for $\varphi$ in all positions of $\pi$ with respect to the residual schedulers $\mathcal{S} \uparrow \pi_{[0..i]}$ (see Section II for the definition of these):

$$lrp_{\varphi}^\mathcal{S}(\pi) = \liminf_{n \to \infty} \frac{1}{n+1} \cdot \sum_{i=0}^{n} Pr_{M, \pi_{[i..]}}^\mathcal{S}(\varphi).$$

The long-run probability for property $\varphi$ under scheduler $\mathcal{S}$ from state $s$, denoted $LP_{M,s}^\mathcal{S}(\varphi)$, is defined as the expectation of the random variable $\pi \mapsto lrp_{\varphi}^\mathcal{S}(\pi)$ under $\mathcal{S}$ starting at state $s$. We now address the task to compute the extremal long-run probabilities for $\varphi$:

$$LP_{M,s}^{\max}(\varphi) = \sup_{\mathcal{S}} LP_{M,s}^\mathcal{S}(\varphi)$$

$$LP_{M,s}^{\min}(\varphi) = \inf_{\mathcal{S}} LP_{M,s}^\mathcal{S}(\varphi)$$

where $\mathcal{S}$ ranges over all schedulers for $M$, and refer to this task as the lrp-problem. In contrast to classical optimization problems for MDPs, the random variable whose expectation we aim to optimize, namely $lrp_{\varphi}^\mathcal{S}$, depends on the scheduler $\mathcal{S}$.

**Remark IV.1 (Long-run frequencies and probabilities).** Computing extremal long-run frequencies for path properties in KSs is a special case of the lrp-problem: KSs can be viewed as MDPs where all distributions are Dirac. For such degenerated MDPs, if $\pi$ is an infinite $\mathcal{S}$-path then $Pr_{M, \pi_{[0..]}}^\mathcal{S}(\varphi) = \prod_{i\in\mathbb{N}} \mathcal{S}(\varphi)$.

Hence, $lrp_{\varphi}(\pi) = lrp_{\varphi}^\mathcal{S}(\pi)$, and therefore $LP_{M,s}^{\max}(\varphi) = \sum_{s'} \mathcal{S}(s') \cdot LP_{M,s'}^{\max}(\varphi)$ and $LP_{M,s}^{\min}(\varphi) = \sum_{s'} \mathcal{S}(s') \cdot LP_{M,s'}^{\min}(\varphi)$.

Analogously to extremal long-run frequencies in KSs, we can assume MDPs to be strongly connected. Indeed for a general MDP $M$ without traps, one can compute the MECs $\epsilon_1, \ldots, \epsilon_k$ in polynomial-time [13], [19].

Given a path formula $\varphi$, we define a weighted MDP $N$, obtained from $M$ by collapsing each MEC $\epsilon_i$ into a single state, say $s_i$, which is duplicated into $s_i'$ a trap state. The copy $s_i'$ is reachable only from $s_i$ by a Dirac action on a fresh action symbol. The weight function assigns $LP_{\mathcal{S}}^{\max}(\varphi)$ to state $s_i'$ and 0 to all other states. With this construction, $LP_{M,s}^{\max}(\varphi)$ equals the maximal expected accumulated weight in $N$ under all proper schedulers, i.e., schedulers that reach the trap states almost surely. This value is computable in polynomial-time using standard algorithms for the stochastic shortest path problem [1], [4], [21].

When $M$ is strongly connected, the optimal long-run probabilities do not depend on the starting state and we simply write $LP_{M}^{\max}(\varphi)$ and $LP_{M}^{\min}(\varphi)$.

**A. Efficiently solvable lrp instances**

We first investigate special cases for which one can obtain efficient algorithms to compute optimal long-run probabilities: first, we explain the case of Markov chains, and then we identify restricted classes of properties for MDPs.

**Remark IV.2 (Special case of Markov chains).** A Markov chain can be seen as a degenerated MDP where each state has a unique enabled action, in which case the concept of schedulers is irrelevant and we simply write $lrp_{\varphi}(\pi)$ for each infinite path $\pi$. If $\varphi$ is an $\omega$-regular property, for each bottom strongly connected component (BSCC) $B$ of the Markov chain $M$, the long-run probability for all states in $B$ is the same:

$$LP_B(\varphi) = \sum_{t \in B} lrt_B(t) \cdot Pr_{M,t}(\varphi)$$

where $lrt_B(t)$ denotes the steady-state probability (defined as the long-run frequency) of state $t$ in $B$. Thus, $LP_B(\varphi)$ equals the probability for $\varphi$ in $B$ viewed as a Markov chain where the initial distribution is given by the long-run frequencies in $B$, which again coincides with the expected mean payoff in $B$ when $Pr_{M,t}(\varphi)$ is viewed as weight for state $t$. The long-run frequencies inside the BSCC are computable in exponential time using a linear equation system. The values $Pr_{M,t}(\varphi)$ for the states inside the BSCC are computable using standard techniques for the analysis of Markov chains against $\omega$-regular properties (see e.g. [3]). The complexity depends on the type and representation of $\varphi$: for instance, exponential-time algorithms exist for LTL formulas [18]. Thus, long-run probabilities for LTL-properties in Markov chains are computable in exponential time. Moreover, $LP_{M,s}(\varphi)$ is computable in polynomial time for those properties $\varphi$ where $Pr_{M,t}(\varphi)$ is computable in polynomial time, such as until, Rabin or Streett properties.

Alternatively, the computation of long-run probabilities as expected mean payoff when dealing with the weight function that assigns weight $wgt(s) = Pr_{B,s}(\varphi)$ to each state $s$ can also be written as a quotient of expectations as follows. Let $s$ be an arbitrary state in $B$, called reference state. Then, the long-run probability for $\varphi$ in $B$ equals the quotient of the expected accumulated weight along paths of length at least 1 from $s$ until returning to $s$ and the expected return time (i.e., expected number of steps) along such paths from $s$ to $s$. (Strong connectivity ensures that both expectations are finite.) That is,

$$LP_B(\varphi) = \frac{E_{B,s}(\text{"weight until } s\text{"})}{E_{B,s}(\text{"steps until } s\text{"})} \quad (\ast)$$

Finally, if $B$ denotes the set of all BSCCs of $M$ then
for each state $s$ in $M$:

$$\mathbb{LP}_{M,s}(\varphi) = \sum_{\pi \in \mathcal{B}} \Pr_{M,s}(\diamond \varphi_B) \cdot \mathbb{LP}_{\mathcal{B}}(\varphi).$$

We now identify classes of path formulas, for which the lrp-problem is solvable in polynomial time. This applies to (repeated) reachability and other properties $\varphi$ where efficiently computable schedulers $\mathcal{S}$ exist that maximize the probabilities for $\varphi$ from every visited state in the sense that

$$\Pr_{M,\pi,\mathcal{S}[i]}(\varphi) = \Pr_{M,\pi,\mathcal{S}[i]}(\varphi)$$

for each infinite $\mathcal{S}$-path $\pi$ and each position $i \in \mathbb{N}$. Obviously, such a scheduler $\mathcal{S}$ maximizes the long-run probability for $\varphi$. (See Theorem IV.3 and its proof.)

**Theorem IV.3 (Efficiently solvable lrp-instances).** The values $\mathbb{LP}_{\mathcal{S}[\mathcal{A}]}(\varphi)$ and $\mathbb{LP}_{\mathcal{S}[\mathcal{B}]}(\varphi)$ are computable in polynomial-time if $\varphi$ has the form $\diamond \varphi$ (reachability), $\Box \varphi$ (invariant), $\bigwedge_{i=1}^{n} \bigvee_{i=1}^{n} (\Box \varphi_{B_{i1}} \land \Box \varphi_{C_{i1}})$ (generalized Rabin condition), or $\bigwedge_{i=1}^{n} (\Box \varphi_{a_{i1}} \rightarrow \Box \varphi_{b_{i1}})$ (Streett condition). Moreover, FMD-schedulers are optimal for all these cases, and even MD-schedulers for reachability, invariant, Büchi and co-Büchi conditions.

An analogous result for much richer classes of properties cannot be expected given that, already in the non-probabilistic setting, infinite-memory can be necessary for until properties (see Fig. 1 left), and co-safety properties yield PSPACE-hardness (see Theorem III.7).

**B. Long-run probabilities for until properties**

We now address the case of an until property $a \mathcal{U} b$. While long-run frequencies for $a \mathcal{U} b$ in KSs can be handled fairly easily using a reduction to classical mean payoff objectives, the probabilistic setting adds major extra challenges. Recall that in strongly connected KSs, if no infinite-memory is required, then optimal long-run frequencies can be achieved by strategies operating in just two modes. This stands in contrast with the probabilistic setting where, in those cases where no infinite-memory is needed, optimal schedulers can need a counter for the number of consecutive $a$-states up to some bound that depends on $M$ (rather than a constant). This phenomenon is illustrated in the example from Fig. 2, with which we carry on now.

**Example IV.4.** For the MDP $\mathcal{N}_k$ from in Fig. 2, we already argued that the two MD-schedulers $\mathcal{S}_{\alpha}$ and $\mathcal{S}_{\beta}$ achieve long-run probabilities of $1/2$ and $\frac{1}{\sqrt{k}^2}$ for the until property $a \mathcal{U} b$. Thus, as soon as $k \geq 3$, $\mathcal{S}_{\alpha}$ achieves the higher long-run probability, whereas $\mathcal{S}_{\beta}$ maximizes the probability of $a \mathcal{U} b$ from the $a$-state. However, none of them is optimal. To see this, consider the FMD-schedulers $\mathcal{T}_n$ for $n \geq 1$, that use a counter for the number of consecutive visits to the $a$-state, starting with counter value 1 when entering that state via the transitions from the other states. When in the $a$-state, $\mathcal{T}_n$ schedules action $\alpha$ if the counter value is at most $n$ and $\beta$ otherwise.

We compute $\mathbb{LP}_{\mathcal{T}_n}(a \mathcal{U} b)$ via the quotient representation shown in (s) in Remark IV.2 using that the Markov chain $\mathcal{C}_n$ induced by $\mathcal{T}_n$ is strongly connected and consists of the states $b_1, c_1, b_2, c_2$ and the states $(a, 1), (a, 2), \ldots, (a, n+1)$, where $(a, 1)$ means state $a$ with counter value 1. We pick $(a, 1)$ as reference state.

Let us first compute the denominator. The expected return time from $(a, 1)$ to $(a, 1)$ can be written as the sum of the expected frequencies $e_f(s)$ of the states $s$ in $\mathcal{C}_n$ along the return paths from and to $(a, 1)$. These values are: $e_f(a, i) = \frac{1}{3^i}$ for $i = 1, \ldots, n+1$, $e_f(b_1) = e_f(c_1) = (1 - \frac{1}{3^n}) \cdot \frac{1}{2}$, $e_f(b_2) = \frac{1}{3^n}$, and $e_f(c_2) = \frac{1}{3^n}$. Note that each of the states $(a, i), b_1, c_1$ and $b_2$ occurs exactly once on each return path from $(a, 1)$ to $(a, 1)$. Thus, for these states $s$, the expected frequency equals the probability of reaching $s$ from the reference state $(a, 1)$. For state $c_2$, we take into account that the self-loop is taken an expected $k - 1$-times. We conclude:

$$\mathbb{E}_{\mathcal{C}_n}(e_f(a, i)) \text{"steps until } (a, 1)\" = \sum_{i=1}^{n+1} \frac{1}{3^i} + (1 - \frac{1}{3^n}) \cdot \frac{1}{2} + \frac{1}{3^n} + \frac{k}{3^n} = \frac{1}{4} \cdot (10 + (4k - 2) \cdot \frac{1}{3^n})$$

We now compute the expected accumulated weight along the return paths from and to $(a, 1)$ under scheduler $\mathcal{T}_n$. This value can be computed as the sum of the expected frequencies of every state $s$ multiplied with its weight. In our case, the weights are the probabilities for $a \mathcal{U} b$ in $\mathcal{C}_n$. That is:

$$\mathbb{E}_{\mathcal{C}_n}(e_f(a, i)) \text{"weight until } (a, 1)\" = \sum_s e_f(s) \cdot \Pr(e_f(a, i))$$

where $s$ ranges over all states in the Markov chain $\mathcal{C}_n$ induced by $\mathcal{T}_n$. The probability values are as follows: $\Pr(e_f(a, i)) = \frac{1}{2} (1 + \frac{1}{3^{i+1}})$ for $i = 1, \ldots, n+1$, $\Pr(e_f(b_1)) = \Pr(e_f(b_2)) = 1$, and $\Pr(e_f(c_1)) = \Pr(e_f(c_2)) = 0$. So, we get:

$$\mathbb{E}_{\mathcal{C}_n}(e_f(a, i)) \text{"weight until } (a, 1)\" = \sum_{i=1}^{n+1} \frac{1}{3^i} \cdot \frac{1}{2} (1 + \frac{1}{3^{n+1}}) + (1 - \frac{1}{3^n}) \cdot \frac{1}{2} - 1 = \frac{1}{4} \cdot (5 + (2n+3) \cdot \frac{1}{3^n})$$

$$\mathbb{LP}_{\mathcal{T}_n}(a \mathcal{U} b) = \frac{5 + (2n+3) \cdot \frac{1}{3^n}}{10 + (4k - 2) \cdot \frac{1}{3^n}}.$$
Thus, $\Xi_n$ achieves a higher long-run probability than $\mathcal{E}_k$ if $n > k - 2$. To determine which scheduler is optimal among the schedulers $\Xi_n$ with $n \in \mathbb{N}$, we determine the least natural number $n$ such that $\bigcap_{k=0}^{n-1} (a \cup b) > \bigcap_{k=0}^{n} (a \cup b)$. Treating the computed expression for $\bigcap_{k=0}^{n} (a \cup b)$ as a real function in $n$, one can check that the derivative has only one root and that hence the obtained value $n$ indeed yields the optimal scheduler among these schedulers. For $k \geq 2$, we obtain $n = k - 1$.

We will see later that the maximal long-run probability of $\mathcal{N}_k$ is indeed achieved by $\Xi_n$ for this $n$ (see Remark IV.12 below). Note that $\mathcal{N}_k$ has 5 states and its size is in $O(\log k)$. So, this example illustrates that even in cases where finite memory is sufficient, the memory requirements of optimal schedulers can grow exponentially with the size of the MDP. The same applies to the logarithmic length of the optimal values.

To see this, we observe that in

$$\bigcap_{k=0}^{n} (a \cup b) = 5 \cdot 3k^{-1} + 2k + 1$$

$$10 \cdot 3k^{-1} + 4k - 2$$

the greatest common divisor of numerator and denominator is at most 4 (note that $2(5 \cdot 3k^{-1} + 2k + 1) - (10 \cdot 3k^{-1} + 4k - 2) = 4$). Therefore, the binary representation of the optimal value requires exponentially many bits in the size of $\mathcal{N}_k$.

After this example that highlighted the challenge in computing optimal long-run probabilities, in the remainder of this section, we provide a pseudo-polynomial time algorithm for computing extremal long-run probabilities for $a \cup b$. We first fix useful notations. Given a state $s$ of $\mathcal{M}$, let $p^\text{max}_s = p^\text{max}_{\mathcal{E}_s}(a \cup b)$, $p^\text{min}_s = p^\text{min}_{\mathcal{E}_s}(a \cup b)$, and:

$$A = \{s \in S \mid p^\text{max}_s > 0$ and $p^\text{min}_s < 1\}$$

$$B = \{s \in S \mid p^\text{min}_s = 1\}$$

$$C = S \setminus (A \cup B)$$

Then, $p^\text{opt}_{\mathcal{E}_s}(a \cup b) = p^\text{opt}_{\mathcal{E}_s}(A \cup B)$ for every $s$ and $\mathcal{E}$. Hence, we may safely assume that the labeling function fulfills $a \in L(s)$ iff $s \in A$ and $b \in L(s)$ iff $s \in B$. For $T \subseteq S$, a $T$-EC denotes an end component consisting of $T$-states.

Let us first consider the four qualitative variants of the lrp-problem where the objective is to decide the existence of a scheduler $\mathcal{E}$ such that $\bigcap_{k=0}^{n} (a \cup b)$ equals 1, 0, is strictly less than 1 or is positive.

**Lemma IV.5 (Qualitative lrp-problems).** The qualitative variants of the lrp-problem for MDPs and until properties are decidable in PTIME.

**Proof sketch.** Positivity check is trivial as (in strongly connected MDPs): $\bigcap_{k=0}^{n} (a \cup b) > 0$ for some scheduler $\mathcal{E}$ iff $B \neq \emptyset$. For the value 1 problem, we can rely on: $\bigcap_{k=0}^{n} (a \cup b) = 1$ iff $\exists \mathcal{E} \exists_{k=0}^{n} (a \cup b) = 1$ iff $\mathcal{M}$ has a $A \cup B \cup \mathcal{E}$ with $\mathcal{E} \cap B = \emptyset$ or $\mathcal{M}$ has an $A \cup B \cup \mathcal{E}$ with $\mathcal{E} \cap B = \emptyset$. The arguments for the other two problems are analogous. For details see Lemma A.10.

We observe that we may concentrate on maximal long-run probabilities for until-properties since (see Lemma A.11):

$$\bigcap_{k=0}^{n} (a \cup b) = 1 - \bigcap_{k=0}^{n} (a \cup b)$$

if $\mathcal{M}$ has no $A \cup B \cup \mathcal{E}$. Otherwise $\bigcap_{k=0}^{n} (a \cup b) = 0$.

For the rest of this section, we suppose that $\mathcal{M}$ is strongly connected and does not have $A \cup B \cup \mathcal{E}$ with $\mathcal{E} \cap B = \emptyset$, as otherwise $\bigcap_{k=0}^{n} (a \cup b)$ is 1.

The first important result is that $\bigcap_{k=0}^{n} (a \cup b)$ can be approximated by finite-memory schedulers. This is proved using Fatou’s lemma (see Lemma A.12).

**Lemma IV.6.** $\bigcap_{k=0}^{n} (a \cup b) = \sup_{\mathcal{E} \in \text{FM}} \bigcap_{k=0}^{n} (a \cup b)$.

The main idea behind our algorithm is that particular FM-schedulers suffice to achieve the extremal long-run probabilities. In fact, we prove the existence of a saturation point $K \in \mathbb{N}$, such that the only kind of memory schedulers need is to count the number of consecutive $a$-states up to the current instant. We establish this by proving that each FM-scheduler $\mathcal{E}$ can be “improved” into another FM-scheduler if it does not maximize the probability for $a \cup b$ (which does not require memory) after having generated a sequence of $K$ consecutive $a$-states. Here, improving means increasing the long-run probability of $a \cup b$. We will show that such a saturation point $K$ is computable in time polynomial in the size of $\mathcal{M}$.

For $\alpha \in \text{Act}(s)$, let $p^\text{opt}_{\alpha}(a \cup b) = \sum_{t \in S} p(s, \alpha, t) \cdot p^\text{max}_t$ and we write $\text{Act}^\text{opt}(s)$ for the set of maximizing actions, i.e. actions $\alpha \in \text{Act}(s)$ with $p^\text{opt}_{\alpha}(a \cup b) = p^\text{max}_s$.

For a given bound $K$, define the class FM($K$) of FM-schedulers $\mathcal{E}$ for $\mathcal{M}$ such that

1) $\text{Pr}_{\mathcal{E}_s}(a \cup b) = \text{Pr}_{\mathcal{E}_s}(a \cup b)$ for each finite $\mathcal{E}$-path $\alpha$ that has a suffix consisting of $K$ or more consecutive $a$-states, and

2) the Markov chain induced by $\mathcal{E}$ has a single BSCC.

By definition, any $\mathcal{E} \in \text{FM}(K)$ only schedules maximizing actions (in $\text{Act}^\text{opt}(\cdot)$) for paths ending with $K$ or more consecutive $a$-states (by 1); moreover, all states in $\mathcal{M}$ have the same long-run probability under $\mathcal{E}$, written $\bigcap_{k=0}^{n} (a \cup b)$, (by 2). We now strengthen the statement of Lemma IV.6 as follows:
Lemma IV.7 (Saturation point). There exists $K$ computable in polynomial time and satisfying:

$$L^\text{max}_{\mathcal{M}}(aUb) = \sup_{\Theta \in \text{FM}(K)} L^\Theta_{\mathcal{M}}(aUb).$$

Proof sketch. Let us explain the choice of $K$ and its computation. For the soundness see Lemma A.13. Let

$$\delta = \max \left\{ \left\lvert A_i \right\rvert, \left\lceil \frac{e}{\delta} \right\rceil \right\} \quad (*)$$

where $\delta$ and $e$ are defined as follows. Let

$$\delta = \min \left\{ \frac{p^\text{max}_s - p_s,\alpha}{\alpha \in \text{Act}(s) \setminus \text{Act}^\text{max}(s)} \right\}$$

with the convention that $\delta = \infty$. The value $\delta$ is then set as $\delta = \min_{s \in A} \delta_s$ if there exists at least one $A$-state $s$ with $\delta_s < \infty$ and $\delta = 1$ otherwise. Obviously, $\delta$ is computable in polynomial time in the size of $M$.

Intuitively, each $\delta_s$ is the minimum probability loss when the first action of an optimal strategy for $aUb$ is replaced by a non-optimal action.

Let us now give a construction which allows us to define $e$ and also to compute long-run probabilities under schedulers of FM($K$). Consider an MDP which arises from $M$ by adding a counter representing the number of consecutive visits to $A$-states. When the counter value exceeds a given threshold, only maximizing actions (in $\text{Act}^\text{max}(\cdot)$) are enabled. Formally, given a positive integer $n$, we define the MDP $M_n = (S_n, \text{Act}, p_n)$ as follows. $S_n = B \cup C \cup (A \times \{1, \ldots, n, \top\})$, and the transition probability function is defined by:

- If $s \in B \cup C$ then:
  $$p_n(s, \alpha, (s', 1)) = p(s, \alpha, s') \quad \text{if } s' \in A$$
  $$p_n(s, \alpha, (s', i)) = 1 \quad \text{if } s' \in B \cup C$$

- If $(s, i) \in A \times \{1, \ldots, n-1\}$ then:
  $$p_n((s, i), \alpha, s') = p(s, \alpha, s') \quad \text{if } s' \in B \cup C$$
  $$p_n((s, i), \alpha, (s', i+1)) = 1 \quad \text{if } s' \in A$$

- If $s \in A$, $k \in \{n, \top\}$ and $\alpha \in \text{Act}^\text{max}(s)$ then:
  $$p_n((s, k), \alpha, s') = p(s, \alpha, s') \quad \text{if } s' \in B \cup C$$
  $$p_n((s, k), \alpha, (s', \top)) = p(s, \alpha, s') \quad \text{if } s' \in A$$
  $$p_n(\top) = 0 \quad \text{in all remaining cases.}$$

For every state $s \in S$ of $M_n$, let $s_n$ denote the “corresponding” state in $M_n$. That is, $s_n = s$ if $s \in B \cup C$. For $s \in A$, we pick an arbitrary state $(s, i)$ in $M_n$ that is reachable from $B \cup C$ in $M_n$ and put $s_n = (s, i)$. In the following, we identify $M_n$ with its fragment consisting of the states reachable from the set $\{s_n : s \in S\}$. As $M_n$ is strongly connected, provided that $n \geq |A|$, Letting $N = |A|$, we define

$$e_{t,s} = E_{M_n,t,s}^\text{min}(\text{“steps until } s_n\text{”})$$

as the minimal expected number of steps from state $t_n$ to $s_n$ in $M_n$, and define $e = \max_{s \in S} e_{t,s}$. The values $e_{t,s}$ and a corresponding MD-scheduler $\mathcal{R}_{N,s}$ for $M_n$ that minimizes the expected number of steps to state $s_n$ from every state $t_n$ are computable in polynomial-time using standard linear programming techniques for stochastic shortest paths [4], [21]. The size of $M_n$ is at most quadratic in the size of $M$.

Intuitively, $K$ is chosen such that after a path of $K$ consecutive $a$-states to $s$, a scheduler choosing $\beta \notin \text{Act}^\text{max}(s)$ can be improved by instead choosing an action in $\text{Act}^\text{max}(s)$ and hence increasing the probability to satisfy $aUb$ in the $K$-many $a$-states and afterwards returning to $s$ via a memoryless scheduler.

We define the MDP $\mathcal{K} = M_K$ extended by the following transition-based weight function:

- transitions from $B$-states have weight 1,
- transitions from states in $A \times \{k\}$ with $k \in \{1, \ldots, K-1\}$ leading to a $B$-state have weight $k$,
- transitions from states $(s, K)$ where $s \in A$ have weight $K \cdot p^\text{max}_s$.
- transitions from states $(s, \top)$ where $s \in A$ have weight $p^\text{max}_s$, and
- the weight of all other transitions is 0.

We reduce the computation of maximal long-run probabilities in $M$ to computing the maximal expected mean payoff in $\mathcal{K}$ (see also Corollary A.18):

**Lemma IV.8.** $L^\text{max}_{\mathcal{K}}(aUb) = E^\text{max}_{\mathcal{K}}(MP)$. 

Proof sketch. Clearly, all schedulers for $\mathcal{K}$ correspond to schedulers for $M$. Vice versa, the behavior of FM($K$)-schedulers for $M$ can be mimicked by schedulers for $\mathcal{K}$. Thanks to Lemma IV.7, we deduce that $L^\text{max}_{\mathcal{M}}(aUb) = L^\text{max}_{\mathcal{K}}(aUb)$.

Now, for FM-schedulers $\mathcal{S}$ in $\mathcal{K}$ that have no BSCC consisting of $A$-states, the expected mean payoff agrees with the long-run probability for $aUb$, while otherwise $L^\text{max}_{\mathcal{K}}(aUb) \leq L^\text{max}_{\mathcal{K}}(MP)$. Also, each FM-scheduler $\mathcal{S}$ in $\mathcal{K}$ that has a BSCC consisting of $A$-states can be transformed into an infinite-memory scheduler $\mathcal{E}$ for $\mathcal{K}$ such that $E^\mathcal{K}_{\mathcal{K}}(MP) \leq L^\mathcal{K}_{\mathcal{K}}(MP)$. Therefore $E^\text{max}_{\mathcal{K}}(MP) = L^\text{max}_{\mathcal{K}}(aUb)$. 

**Example IV.9.** Consider the MDP $\mathcal{N}_k$ with $k = 2$ in Example IV.4 which is depicted in Fig. 2. We call this MDP $\mathcal{M}$ and use the notation $M_n$ as well as the other notations as above. First, we compute the saturation point $K$: $\delta = p^\text{max}_{a} - p_{a,\alpha} = \frac{1}{2}$. For the computation of $e$, we consider the MDP $M_1$. As we only consider the states reachable from $(a, 1)$ and only action $\beta$ is available in this state in $M_1$, we see that $e$ equals the expected number of steps from the left $b$ state to $(a, 1)$. So, $e = k + 1 = 3$. Therefore, the saturation point is given by $K = \frac{e}{\delta} = 9$. We obtain the MDP $\mathcal{K} = M_9$ extended with the weight structure described above.
For $0 \leq n \leq 8$, let $\mathcal{T}_n$ be the scheduler which chooses $\alpha$ in state $(a,i)$ if $i \leq n$ and $\beta$ otherwise. All MD-schedulers for $\mathcal{K}$ are of this form. By summation over all paths from $(a,1)$ to $(a,1)$, we compute

$$E^n_{\mathcal{K}}(\text{MIP}) = \frac{\sum_{l=1}^{n} \frac{n}{2l}(n+1) + \frac{n}{2l}(n+2)}{2\sum_{l=1}^{n} \frac{1}{2l}(n+1) + \frac{1}{2l}(n+3)} = \frac{5n+6}{10+n/2}.$$  

The maximum is obtained for $n = 1$. As some MD-scheduler maximizes the expected mean payoff in $\mathcal{K}$, we conclude that $E^{\max}_{\mathcal{K}}(\text{MIP}) = E^{\max}_{\mathcal{K}}(\text{MIP}) = \frac{5}{9}$. This value matches the value $\text{LP}^{\max}_{\mathcal{K}}(a \cup b)$ computed in Example IV.4.

The length of the binary representation of $K$ is polynomial in the size of $M$, so that the size of $\mathcal{K} = M_K$ is pseudo-polynomial in the size of $M$. We thus derive a complexity upper-bound for the computation of optimal long-run probabilities of until properties:

**Theorem IV.10 (Computing optimal values).** The values $\text{LP}^{\max}_{\mathcal{M}}(a \cup b)$ and $\text{LP}^{\min}_{\mathcal{M}}(a \cup b)$ are computable in pseudo-polynomial time.

Example IV.4 illustrates that optimal schedulers can need a counter for the number of consecutive $a$-states up to a saturation point that can grow exponentially in the size of the MDP. Moreover, the logarithmic length of the maximal long-run probability can also be exponential in size of the MDP. This indicates that one cannot expect a polynomial-time algorithm to compute optimal schedulers or their values. Even the threshold problem, asking whether $\text{LP}^{\max}_{\mathcal{M}}(a \cup b) \geq \delta$, is hard:

**Theorem IV.11.** The threshold problem for Until properties in MDP is NP-hard.

**Proof sketch.** We sketch the main idea here. For the full proof see Theorem A.19. We provide a polynomial reduction from the intersection problem for unary DFA, i.e., DFA over a one-letter alphabet. This problem is known to be NP-complete [5]. Let $D_1, \ldots, D_k$ be unary DFAs, and let $\ell$ be the product of their numbers of states. Then, $L(D_1) \cap \ldots \cap L(D_k)$ is nonempty if and only if $L(D_1) \cap \ldots \cap L(D_k)$ contains a word of length at most $\ell$.

We construct an MDP $M$ with the disjoint union $A$ of these DFAs as a substructure and a threshold $\delta$ such that a scheduler has to attempt to follow a path of the same length to an accepting state in each of the DFAs in order to surpass the threshold. The states in $M$ are the $A$-states and four additional states init, a, b, c. States in $A$ and $a$ are labeled with $a$ and $b$ is labeled with $b$. In init and $a$, actions pump leading to $a$ and enter randomly leading to one of the initial states in $A$ are available. In each state in $A$ an action $\alpha$ following

the transition in the DFAs is available. All of these actions may fail with a small probability of $\frac{1}{q}$. Failure leads back to init. In the final states in $A$, an action $\beta$ leading to $b$ is available. In $b$ and $c$ an action is available which leads to init with probability $\frac{1}{q}$ and to $c$ otherwise.

The values $r, r', \delta$ are chosen such that a scheduler $\mathcal{S}$ achieves $\text{LP}^{\max}_{M}(a \cup b) \geq \delta$ if and only if it chooses $\beta$ exactly in the moment it visited $\ell$ consecutive $a$-states. For this, $\mathcal{S}$ can pump $m$ times before entering $A$, but then has to follow the transitions in a randomly chosen DFA $D_i$ for exactly $\ell-m$ steps before choosing $\beta$. But this is only possible if there is a word of length $\ell$ which is accepted by all the DFAs. By the definition of $\ell$, this is in turn equivalent to the non-emptiness of the intersection $\bigcap_i L(D_i)$. The main difficulty is to make sure that the chosen $\delta$ is computable in polynomial time.

**Remark IV.12 (Memory requirements).** If $M$ does not have an $A$-EC then there are FMD-schedulers achieving the optimal long-run frequency for $a \cup b$ that behave memoryless in $(B \cup C)$-states and that use a counter for the number of consecutive $A$-states up to the saturation point. For MDPs with $A$-EC, optimal schedulers can need infinite-memory depending on whether all MD-schedulers for $K$ with maximal expected mean payoff have a BSCC consisting of states in $A \times \{\top\}$. This, e.g., applies to cases where each EC of $M$ contains some $B$-state also contains a $C$-state and there is an $A$-EC $E$ where $\text{LP}^{max}_{E}(a \cup b) > p^{\max}_{s}$ for all $A$-states $s$ that do not belong to some $A$-EC.

**Proposition IV.13 (Efficiently solvable cases).** $\text{LP}^{\max}_{M}(a \cup b)$ can be computed in polynomial time in each of the following particular cases

- if $M$ has no $A$-cycle, or
- if for every scheduler $\mathcal{S}$ and every $A$-state $s$, $\text{LP}^{\max}_{M,s}(a \cup b) = p^{\max}_{s}$.

**Proof.** In the first case indeed, we can let $K = |A|$ as saturation point, in which case the size of $K = M_K$ is only polynomial in the size of $M$. In the second case, $\text{LP}^{\max}_{M}(a \cup b)$ is simply the maximal mean payoff in $M$ when the weight of state $s$ is $p^{\max}_{s}$.

**V. Conclusion**

The results of the paper illustrate that the introduced notions of long-run frequencies and probabilities for reasoning about (long-run) average satisfaction of path properties lead to computable measures in various cases. While efficient and simple algorithms exist for reachability, invariance, Streett and Rabin properties, the complexity raises to PSPACE for co-safety. The
probabilistic setting adds extra difficulties as illustrated for the case of until properties in MDPs.

There are various interesting open problems. The major open problem in the non-probabilistic setting is the computability of long-run frequencies in KSs for \( \omega \)-regular properties specified by a Büchi automaton. Future directions in the probabilistic setting include the precise complexity of the threshold problems for until properties and algorithms for computing long-run probabilities for co-safety or \( \omega \)-regular properties and long-run expectations of, e.g., accumulated or discounted weights.

REFERENCES


Appendix

A. Long-run frequencies for until properties in KS

Lemma A.1 (see Lemma III.1).

(a) The following three statements are equivalent:
   (1) $\mathbb{L}_{\mathcal{T}}\text{max}(A \cup B) = 1$
   (2) $\mathcal{T}$ has an infinite path $\pi$ with $\text{lr}\mathcal{I}_{A \cup B}(\pi) = 1.$
   (3) $\mathcal{T}$ has an $A \cup B$-cycle.

(b) The following three statements are equivalent:
   (1) $\mathbb{L}_{\mathcal{T}}\text{min}(A \cup B) = 0$
   (2) $\mathcal{T}$ has an infinite path $\pi$ with $\text{lr}\mathcal{I}_{A \cup B}(\pi) = 0.$
   (3) $\mathcal{T}$ has an $A \cup B$-cycle.

Proof. We only prove the equivalence of (1), (2) and (3) for maximal long-run frequencies (item (a)).

   “(1) $\implies$ (3)” suppose that $\mathcal{T}$ does not have $A \cup B$-cycles. Then, each cycle in $\mathcal{T}$ contains a state that is not contained in $A \cup B$. If $n = |S|$ is the total number of states then along every path later after $n$ steps a state in $S \setminus (A \cup B)$ will be reached. Hence, the long-run frequency of each infinite path is less or equal $(n-1)/n$. But then $\mathbb{L}_{\mathcal{T}}\text{min}(A \cup B) \leq (n-1)/n < 1$.

   “(3) $\implies$ (2)” suppose that $\mathcal{T}$ has an cycle $\xi$, consisting of states in $A \cup B$. If one of the states in $\xi$ is a $B$-state then the long-run frequency of the infinite path $\xi^\omega$ is 1. Suppose now that all states in $\xi$ are in $A$. Let $\xi = s_0 s_1 \ldots s_n$ and let $\omega = t_0 t_1 \ldots t_m t_{m+1} \ldots t_k$ be a path with $t_0 = s_0 = s_1 = t_k, t_1, \ldots, t_{m-1} \in A, t_m \in B$. (Recall that by definition of $A$ we have $s \models \exists (A \cup B)$ for all states $s \in A$.) Then, the long-run frequency of the infinite path $\xi; \omega; \xi^2; \omega; \xi^4; \omega; \xi^8; \omega; \ldots$ is 1.

   “(2) $\implies$ (3)” obvious.

The argument for statement (b) is analogous and omitted here. (Note that the absence of $A$-cycle yields that $s \models \exists (A \cup C)$ for all states $s \in A$.)

We give the precise definition of the transition function for the weighted KS constructed in Section III-A. We switch from $\mathcal{T}$ to the weighted KS $\mathcal{T}'$ with state space $S' = B \cup C \cup A \times [0,1]$ and the following transitions:

- If $(s,s') \in A$ is a transition in $\mathcal{T}$ where $s \in B \cup C$ then $\mathcal{T}'$ contains the following transitions:
  - $s \rightarrow (s',0)$ and $s \rightarrow (s',1)$ if $s' \in A$
  - $s \rightarrow s'$ if $s' \in B \cup C$

- If $(s,s') \in A$ is a transition in $\mathcal{T}$ where $s \in A$ then $\mathcal{T}'$ has the following transitions:
  - $(s,0) \rightarrow s'$ if $s' \in C$
  - $(s,1) \rightarrow s'$ if $s' \in B$
  - $(s,0) \rightarrow (s',0)$ and $(s,1) \rightarrow (s',1)$ if $s' \in A

Recall that the weight function in $\mathcal{T}'$ is state-based and assigns weight 1 to the states in $B \cup A \times [1]$ and weight 0 to the states in $C \cup A \times [0]$.

We now introduce the following additional notation. For $s \in S$ let $I(s) = \{s\}$ if $s \in B \cup C$ and $I(s) = \{(s,0),(s,1)\}$ if $s \in A$. For the following lemma, the assumption that $\mathcal{T}$ is strongly connected, is irrelevant.

Lemma A.2 (See Lemma III.2). Suppose $\mathcal{T}$ has no $A$-cycles and let $s$ be a state of $\mathcal{T}$. Then, $\mathbb{L}_{\mathcal{T}}\text{max}(A \cup B) = \mathbb{L}_{\mathcal{T}'}\text{max}(A \cup B)$ and $\mathbb{L}_{\mathcal{T}}\text{min}(A \cup B) = \mathbb{L}_{\mathcal{T}'}\text{min}(A \cup B)$.

In particular, if $\mathcal{T}$ is strongly connected then $\mathbb{L}_{\mathcal{T}}\text{max}(A \cup B) = \mathbb{L}_{\mathcal{T}'}\text{max}(A \cup B)$ and $\mathbb{L}_{\mathcal{T}}\text{min}(A \cup B) = \mathbb{L}_{\mathcal{T}'}\text{min}(A \cup B)$ as stated in Lemma III.2.

Proof. We show that there is a bijection $\mathcal{T}$ between the infinite paths in $\mathcal{T}$ and in $\mathcal{T}'$ such that for each infinite path $\pi$ in $\mathcal{T}$: $\mathbb{f}(\pi) \in I(\mathbb{f}(\pi))$ and $\text{lr}\mathcal{I}_{A \cup B}(\pi) = \mathbb{f}(\pi)$.

Given an infinite path $\pi = s_0, s_1, s_2, \ldots$ in $\mathcal{T}$, we define $\mathbb{f}(\pi) = s_0' s_1' s_2' \ldots$ as follows. If $s_i \in B \cup C$ then $s_i' = s_i$. If $i = 0$ and $s_i \in B \cup C$ and $s_i s_{i+1} \ldots s_k \in A$ then $s_i' s_{i+1}' \ldots s_k' = (s_i,1) (s_{i+1},1) \ldots (s_k,1)$. Likewise, if $i = 0$ and $s_i \in B \cup C$ and $s_i s_{i+1} \ldots s_k \in A$ then $s_i' s_{i+1}' \ldots s_k' = (s_i,1) (s_{i+1},0) \ldots (s_k,0)$. It is easy to see that $\mathbb{f}(\pi)$ is indeed a path in $\mathcal{T}'$ satisfying $\mathbb{f}(\pi) \in I(\mathbb{f}(\pi))$ and $\text{lr}\mathcal{I}_{A \cup B}(\pi) = \mathbb{f}(\pi)$.

B. Long-run frequencies for co-safety properties in KS

Soundness of the weighted KS $\mathcal{G}$

Before providing the proof of Theorem III.5, we show that there are more empty tracks to insert tracks at states $H'$ and one at $q_0$ at the second step.

Lemma A.3. For all paths $(s_k, f_k)_{k \geq 0}$ of $\mathcal{G}$, for all $k \geq 0$, the following conditions hold:

- There are at most $\ell$ positions $0 \leq i \leq 2\ell$ such that $f(i) \in Q \times \{true, false\}$.
- There exists $0 \leq i \leq 2\ell$ such that $f_k(i) \in \{q_0\} \times \{true, false\}$.
- If $k > 0$, then for all $0 \leq i < 2\ell$ such that $f_{k-1}(i) = (q, false)$ for some $q \in Q$, for all $q' \in \delta(q, l(s_k))$, there exists $0 \leq j \leq 2\ell$ such that $f_k(j) = (q', false)$.

Proof. We prove the three properties by induction on the length of the paths of $\mathcal{G}$.

Consider any path $(s_k, f_k)_{k \geq 0}$. The properties are true in the initial state $(s_0, f_0)$ by definition. Assume this holds at $(s_k, f_k)$. Consider $(s_{k+1}, f_{k+1})$ such that $\rightarrow_1(s_k+1, f_{k+1}, H) \rightarrow_2(s_k+1, f_{k+1})$.
so that \((s_{k+1}, f'_{k+1}, H)\) is consistent. Observe that by construction, the second and third properties hold whenever there are enough empty tracks in \(f'_k\). In \(f'_k\), all tracks different than \(Q \times \{\text{true}, \text{false}\}\) in \(f_k\) are set to 0; while each track of the form \((q, b) \in Q \times \{\text{true}, \text{false}\}\) is replaced by \((q', b)\) for some successor \(q'\) of \(q\) in \(A\). Thus, \([0 \leq i \leq 2\ell : f'_k(i) \neq 0]\) \(\subseteq \ell\). All properties then follow from this observation. In fact, in \(f_{k+1}\), the track \((q_0, \text{true})\) or \((q_0, \text{false})\) is added and its occurrence is unique since \(q_0\) there is no edge entering \(q_0\) in \(A\). Furthermore, a track \((q, \text{false})\) is added for each \(q \in H\) unless another track with the same state \(q\) already exists. Then duplicate tracks are removed and labeled by merged(·). Thus, no state \(q \in Q\) appears twice in \((s_{k+1}, f_{k+1})\).

\[\square\]

**Theorem A.4 (See Theorem III.5).** Let \(T\) be a strongly connected \(KS, A\) an NFA encoding a co-
 safety property \(\bar{q}\) and \(\bar{s}\) the weighted \(KS\) defined as above. Then, \(L_{\text{FSmax}}(\bar{q}) = \mathcal{M}_{\text{FSmax}}(\Phi)\) and \(L_{\text{FSmin}}(\bar{q}) = \mathcal{M}_{\text{FSmin}}(\Phi)\).

**Proof.** We prove the following even stronger result:

(a) For each infinite path \(\pi\) in \(T\) starting in \(s_0\) there exists an infinite path \(\pi'\) in \(\bar{q}\) with \(\pi' \models \Phi\) and \(l_{\text{FS}}(\pi) = mp(\pi')\).

(b) For each initial infinite path \(\pi'\) in \(\bar{q}\) with \(\pi' \models \Phi\) there is an infinite path \(\pi\) in \(T\) starting in \(s_0\) such that \(l_{\text{FS}}(\pi) = mp(\pi')\).

**Proof of statement (a).** Consider any infinite path \(\pi = (s_k)_{k \geq 0}\) of \(T\) that starts at \(s_0\). We will construct a uniquely defined path \((s_k, f_k)_{k \geq 0}\) in \(\bar{s}\) satisfying \(\Phi\) and whose mean payoff is exactly \(l_{\text{FS}}(\pi)\).

The path follows \(\pi\) in its first component. The only non-determinism in the choice of the second component is in the choice of \((q_0, b)\) with \(b \in \{\text{true}, \text{false}\}\) that is introduced at each step, and in the choice of a successor for each \((q, \text{true})\). In fact, the rest of the components are uniquely determined by the construction.

We are going to construct a path \((s_k, f'_k)_{k \geq 0}\) with an additional information: at each step \(k\), tracks of the form \((q, \text{true})\) are replaced by \((q, \text{true}, \rho)\) where \(\rho\) is a path witnessing that the trace \(L(s_k)\) is accepted by \(A\) from state \(q\). The final path \((s_k, f_k)_{k \geq 0}\) is then obtained by removing this additional component.

We define \(f'_0\) by choosing \((q_0, \text{true}, \rho)\) if and only if \(\rho\) is some witness accepting path of \(A\) from \(q_0\).

Assume that the path is constructed up to step \(k - 1\). Then \(f'_k\) is defined with the following resolution of the non-determinism:

- We add the track \((q_0, \text{true}, \rho)\) if \(\rho\) is a witness path, and add \((q_0, \text{false})\) otherwise.

- For any \(0 \leq i \leq 2\ell\) with \(f'_{k-1}(i) = (q, \text{true}, \rho)\), if \(q \in F\) then \(\rho\) is the empty path and \(f'_k(i) = \emptyset\) (there is no non-determinism to resolve in this case). Otherwise, let us write \(\rho = qq'p'\) where \(q, q'\) are first two states of \(\rho\) and \(p'\) is the rest of the path. We let \(f'_k(i) = (q', \text{true}, q'p')\).

This sequence thus constructed never produces inconsistent states. This is because all tracks are chosen according to the satisfaction of the property \(\bar{q}\) at given position with given witness accepting paths in \(A\).

Now, the mean payoff of \((s_k, f_k)_{k \geq 0}\) is obtained as the average of the number of positions in which a track \((q_0, \text{true})\) is created. This is the case exactly when \(\rho\) holds from a given position. Thus, the mean payoff of \((s_k, f_k)_{k \geq 0}\) is equal to \(l_{\text{FS}}(\pi)\).

It remains to show that \((s_k, f_k)_{k \geq 0}\) satisfies \(\Phi\). Consider any position \(0 \leq i \leq 2\ell\) and any \(k \geq 0\). Let us show that \((\text{false}_1 \lor \text{merged}_1 \lor F_1 \lor \emptyset)\) is satisfied at some position \(l \geq k\). This is clear if \(f_k(i) = \emptyset\lor \text{merged}_1 \lor F_1 \lor \emptyset\), so assume this is not the case. The only remaining case is \(f_k(i) = (q, \text{true})\) for \(q \in Q \setminus F\). Consider the witness path \(\rho\) above. By construction, there is \(l > k\) such that either \(f_l(i) \in F \times \{\text{true}\}\) or \(f_l(i) \in \{\text{merged}\} \times N\).

**Proof of statement (b).** Conversely, consider a path \((s_k, f_k)_{k \geq 0}\) of \(\bar{s}\) satisfying \(\Phi\). Let \(\pi = (s_k)_{k \geq 0}\) be the corresponding path in \(T\). We will prove that for all positions \(k\) such that \(f_k\) contains a track with \((q_0, \text{true})\), \(\pi, k \models \Phi\), and for all other positions \(k, \pi, k \not\models \Phi\). This shows that \(l_{\text{FS}}(\pi)\) is equal to the mean payoff of the path \((s_k, f_k)_{k \geq 0}\).

We prove a more general statement: for all states \((s_k, f_k)\) containing some track \((q, \text{true})\), the suffix \(L(\pi_{[k, \infty]}(s))\) has a prefix that is accepted by some path from \(q\) to \(F\) in \(A\). We proceed by induction on the position \(i\) of the track containing \((q, \text{true})\), with \(0 \leq i \leq 2\ell + 1\).

- Case \(i = 0\). Consider \(k \geq 0\) such that \(f_k(0) = (q, \text{true})\). We claim that there exists \(k' > k\) and states \(q_1 \in Q\) for all \(k \leq l \leq k'\) such that \(f_l(i) = (q_1, \text{true})\), \(q_k = q\) and \(q_k \in F\), and \(q_k, q_{k+1} \ldots q_{k'}\) is a path of \(A\). In fact, since \((s_k, f_k)_{k \geq 0}\) satisfies \(\Phi\), the track \(i\) cannot be of the form \((q', \text{true})\) independently from position \(k\). Moreover, it can never be of the form \((\text{merged}, j)\) since this would imply \(j < 0\). It cannot be either of the form \((q, \text{false})\) before visiting some \((q, \text{true})\) with \(q_f \in F\) since this is not possible in the construction. Thus, in order to satisfy \(\Phi\), there must exist such a position \(k'\) as above. Then \(q_k, q_{k+1} \ldots q_{k'}\) is a path of \(A\) by construction of \(\bar{s}\).

- Consider \(i > 0\) and \(k \geq 0\) such that \(f_k(i) = (q, \text{true})\). We claim that one of the two conditions hold:
1) there exists $k' > k$ and states $q_1 \in Q$ for all $k \leq l \leq k'$ such that $f_l(i) = (q_1, \text{true})$, $q_k = q$ and $q_{k'} \in F$, and $q_k q_{k+1} \ldots q_{k'}$ is a path of $A$.

2) or, there exists $k' > k$ and states $q_1 \in Q$ for all $k \leq l \leq k' - 1$ such that $f_l(i) = (q_1, \text{true})$, $q_k = q$ and $q_1 q_1 \ldots q_{k-1}$ is a path of $A$ and $f_k(i) = (\text{merged}, j)$ with some $j < i$.

The result follows immediately in the first case. Assume the first case does not hold. By $\Phi$, there must exist some position $k' > k$ satisfying $(\text{false}_i \lor \text{merged}_i \lor \text{false}_i \lor \text{true}_i)$.

Consider the least such position $k'$. By construction, we cannot have $(s_k', f_k') \models $false_i or $(s_k', f_k') \models $true_i since all tracks of the form $Q \times \{\text{true}\}$ are stopped by some $F \times \{\text{true}\}$ or merged $\times \mathbb{N}$. The former does not hold by hypothesis so there must exist $j < i$ such that $f_k(j) = (\text{merged}, j)$.

By construction, there exists $q, q' \in Q$ such that $f_{k-1}(i) = (q, \text{true})$ and $f_k(j) = (q, \text{true})$ with $q' \in \delta(q, L(s_{k-1}))$. By induction hypothesis there exists a path $\rho$ from $q'$ to $F$ along a prefix of the trace $L(s_k') \cap L(s_{k'+1})$. Now if $q_k q_{k+1} \ldots q_{k-1}$ is the path of $A$ from the track $i$ from positions $k$ to $k'-1$, then $q_k q_{k+1} \ldots q_{k-1} \cdot \rho$ is an accepting path of $A$ on a prefix of $L(\pi_{[k-1]})$.

\[ \square \]

**Computation of the extremal values**

Let us explain the exponential-time algorithm of Corollary III.6 for computing $L^\text{max}_G(\varphi)$ and $L^\text{min}_G(\varphi)$.

We start by building $G$ which has size $|S|(3G + 1)^{2G + 1}$, which is linear in the size of $T$ and exponential in that of $A$. We then compute the strongly connected components of $G$ and keep those SCCs which contain at least one accepting state per each Büchi condition of $\Phi$. In fact, any infinite path satisfying $\Phi$ must have a suffix that belongs to such an SCC. We then use the algorithm of [28] to compute the cycle that maximizes the mean payoff in time polynomial in the size of $G$.

Then, as done in [6], one can build an infinite path by repeating this cycle, and interleaving an infinite number of visits to accepting states with frequency that goes to 0. A detailed construction of this idea is also given below, in the proof of Theorem A.5.

**PSPACE-completeness of the threshold problem**

We now turn to the proof of Theorem III.7. Let us first recall the result:

**Theorem A.5 (see Theorem III.7).** The threshold problem “given a KS $T$, an NFA $A$ and a rational threshold $\theta$, check whether $T$ has an infinite path $\pi$ with $\text{inf}_{\varphi}(\pi) \geq \theta$” is PSPACE-complete.

The polynomial-space upper bound will be shown in Lemma A.6 and PSPACE-hardness will be shown in Lemma A.7.

**Lemma A.6.** Given KS $T$, co-safety property $\varphi$ described by NFA $A$, and a rational number $\theta$, a comparison operator $\varphi \in \{\geq, >, <, \leq \}$ one can check in polynomial space whether $L^\text{max}_G(\varphi) \geq \theta$.

**Proof.** As PSPACE equals NPSPACE (Savitch’s theorem) it suffices to provide a non-deterministic polynomial-space-bounded procedure to check whether $L^\text{max}_G(\varphi) \geq \theta$. The idea is to use the weighted KS $G$, without constructing $G$ explicitly. To simplify the following argument we concentrate on the case $\geq$.

The algorithm for strict lower bounds and upper bounds are similar.

The algorithm starts by guessing a state $s_0$ and checks whether $s_0$ is reachable from one of the initial states in $I(s_0)$. This check can be done in polynomial space.

Then, it checks whether

1) there exists a cycle $\xi_{\Phi}$ of length $2(|T| + 1)|S_G|$ containing $s_0$ which satisfies each conjunct of $\Phi$,

2) there exists a simple cycle $\xi_{\text{MP}}$ containing $s_0$ with mean payoff of at least $\theta$.

The first check can be done in polynomial space by guessing the path and using a counter up to $2|S_G|$ which can be represented in polynomial space. The second check can be done similarly by keeping the sum of the guessed cycle which can also be represented in polynomial space, since it is bounded by $|S_G|$ as all weights are 0 or 1.

We first show: If the above nondeterministic algorithm returns “yes”, then $G$ has an infinite path $\rho$ with $\text{mp}(\rho) \geq \theta$. This construction follows the more general results on mean payoff parity games of [14]. Let $\xi_{\Phi}$ and $\xi_{\text{MP}}$ denote the two cycles computed above, the first one satisfying $\Phi$, and the second one with mean payoff at least $\theta$. The path we construct first reaches $s_0$. From here, it runs in phases alternating between $\xi_{\Phi}$ and $\xi_{\text{MP}}$. At phase $i \geq 1$, it follows $2^i$ times $\xi_{\text{MP}}$ and then once $\xi_{\Phi}$. This path clearly satisfies $\Phi$ since the cycle $\xi_{\Phi}$ is seen infinitely often. Let $\pi$ denote the suffix of $\rho$ starting at the first occurrence of $s_0$. We calculate the mean payoff of $\pi$, which is the same as the mean payoff of $\rho$. Consider any prefix $\pi_{[0..K]}$ of $\pi$ of length $K$. Let us write $a = |\xi_{\Phi}|$, $b = |\xi_{MP}|$ and $\text{avg}(\pi_{[0..K]})$ for the average payoff of the first $K$ states of $\pi$, i.e.,

$$\text{avg}(\pi_{[0..K]}) = \frac{1}{K} \sum_{i=0}^{K-1} \text{wgt}(\pi_{[i]}).$$
If $K \geq a + 2b$, then there exists a unique integer $i \geq 0$ and some integer $r$ with $0 \leq r < a + 2i + 1$ such that:
\[
K = a + 2b + a + 2b + \ldots + a + 2b + r
\]
\[
= i a + b(2i + 1 - 1) + r
\]
If $r < a + b$, then:
\[
\text{avg}(\pi_{[0..K]}) \geq \frac{\theta(2i + 1 - 1)}{2i + 1 - 1 + (i + 1)a + b}
\]
If $r \geq a + b$, there exists $m > 2i + 1$ such that:
\[
\text{avg}(\pi_{[0..K]}) \geq \frac{\theta(m + 2i + 1 - 1)}{m + 2i + 1 - 1 + i a}
\]
Then, for all $\varepsilon > 0$, there exists $K_\varepsilon$ such that for all $K \geq K_\varepsilon$, $\text{avg}(\pi_{[0..K]}) \geq \theta - \varepsilon$. This yields:
\[
mp(\pi) = \lim_{k \to \infty} \text{avg}(\pi_{[0..K]}) \geq \theta
\]
As the mean payoff of $\rho$ coincides with the mean payoff of each of its suffixes, we get:
\[
mp(\rho) = mp(\pi) \geq \theta
\]
This also shows that the limit of the average payoffs of finite prefixes exists, so the limsup and liminf variants of the mean payoff have also the same value for $\rho$.

Conversely, assume that there exists an infinite path $\rho$ of $\gG$ satisfying $\Phi$, starting in a state of $I(s_0)$ and with $mp(\rho) \geq \theta$. The task is to show that the sketched nondeterministic algorithm has a computation returning the answer “yes”. Pick some state $s_0$ that occurs infinitely in $\rho$. Then, $s_0$ is reachable from $I(s_0)$.

Let us write $\Phi_1$ for the Büchi condition $\square \Diamond (\text{false}_1 \lor \text{merged}_1 \lor G_1 \lor \theta_1)$. Then, $\Phi = \bigwedge_{i=0}^\ell \Phi_i$ and $\rho$ has a fragment that constitutes a cycle in $\gG$ containing state $s_0$ and satisfying all conjuncts $\Phi_i$ of $\Phi$. We now sketch how to construct a new cycles $\xi_\Phi$ and $\xi_{MP}$ satisfying the constraints of (1) and (2). For each $i \in \{0, 1, \ldots, 2\ell\}$, $\rho$ contains a path from $s_0$ to some state $t_i$ with $t_i \models \Phi_i$, and a path from $t_i$ to $s_0$.

A cycle $\xi_i$ of length $\leq 2|s_0|$ containing $s_0$ and $t_i$ can then be constructed by concatenating simple paths from $s_0$ to $t_i$ and from $t_i$ to $s_0$ in $\gG$. Concatenating the cycles $\xi_0, \xi_1, \ldots, \xi_{2\ell}$ for all conjuncts $\Phi_i$ of $\Phi$ yields the desired cycle $\xi_\Phi$. Furthermore, it is known that in mean payoff automata, the maximal mean payoff is achieved on simple cycles [22]. Thus, the algorithm has a computation returning the answer “yes”.

**Lemma A.7.** The threshold problem “given a KS $\gT$, an NFA $A$ and a rational threshold $\theta$, decide whether $\mathbb{L}^\text{mpmax}_\gT(\varphi) \supseteq \theta$” is PSPACE-hard.

**Proof.** The PSPACE lower bound follows by a polynomial reduction from the intersection problem for deterministic finite automata (DFA): given $k$ DFA $D_1, \ldots, D_k$ over the same alphabet $\Sigma$, is the intersection language $L(D_1) \cap \ldots \cap L(D_k)$ nonempty? This problem is known to be PSPACE-complete [29].

To provide a polynomial reduction from the intersection problem for DFA, we suppose we are given DFA $D_1, \ldots, D_k$ over some alphabet $\Sigma$. W.l.o.g. we may assume that $k \geq 2$ and that the empty word is not included in any of the languages $L(D_1)$. Let $Q_1$ be the state space of $D_1$, $t_i = |Q_1|$ and $t = t_1 \ldots t_k$. Then, $L(D_1) \cap \ldots \cap L(D_k)$ is nonempty if and only if there is a word $w \in \Sigma^*$ of length at most $t$ such that $w \in L(D_1)$ for $i = 1, \ldots, k$.

Let $s_1, \ldots, s_k, \#$ be pairwise distinct fresh letters (not contained in $\Sigma$), and let $\Gamma = \Sigma \cup \{s_1, \ldots, s_k, \#\}$.

Given a finite word $w = \sigma_1 \sigma_2 \ldots \sigma_n \in \Sigma^+$, let $\tilde{w}$ denote the word over $\Gamma$ that arises from $w$ by inserting $(k-1)$-times the symbol $\#$ after each letter $\sigma_j$. That is,
\[
\tilde{w} = \sigma_1 \#^{k-1} \sigma_2 \#^{k-1} \ldots \sigma_n \#^{k-1}
\]
For $i = 1, \ldots, k$, one can easily construct in time $O(k^2 + k \cdot \text{size}(D_1))$ a new DFA $B_i$ over the alphabet $\Gamma$ such that:
\[
L(B_i) = \{ s_1 \# t_{i+1} \ldots s_{k-1} \# w : w \in L(D_1), 1 \leq j < k \}
\]
Furthermore, we can construct in time polynomial in $k$ an NFA $A$ over the alphabet $\Gamma$ with:
\[
L(A) = \mathcal{L}(B_1) \cup \ldots \cup \mathcal{L}(B_k) \cup \{\# : i \geq 1\}
\]
Note that $A$ does not accept the empty word and no word starting with a letter in $\Sigma$. Likewise, we can construct in time polynomial in $k$ a strongly connected KS $\gT$ with the following states:

- $s_{i,j}$ for $i = 1, \ldots, k$ and $j = 1, \ldots, k-1$,
- $t_1, \ldots, t_{k-1}$ and
- $u_\sigma$ for each symbol $\sigma \in \Sigma$.

We treat the symbols in $\Gamma$ as atomic propositions and identify the singletons $\{y\}$ with $y$, where $y$ ranges over all symbols of the alphabet $\Gamma$. The labeling function of $\gT$ is then given by:
\[
L(s_{i,j}) = s_i, L(t_j) = \# \text{ and } L(u_\sigma) = \sigma.
\]
The transition relation of $\gT$ is depicted in Figure 4.

Thus, the words generated by $\gT$ are the substrings of the infinite words $y_1 y_2 y_3 \ldots$ where each word $y_i$ has the form
\[
\# s_1^{k-1} \# s_2^{k-1} \ldots \# s_k^{k-1} \tilde{w}
\]
for some $w \in \Sigma^+$. Let
\[
\hat{\theta} = \frac{k(k-1) + (k-1)\ell}{k(k-1) + k\ell}
\]
Clearly, $\gT, A, \hat{\theta}$ can be constructed in time polynomial in the size of the DFA $D_1, \ldots, D_k$. It remains to show that $\gT$ has an infinite path $\pi$ with long-run frequency
Thus, the long-run frequency of the infinite path $\xi$ at least $\theta$ if and only if the intersection language of the $D_i$’s is nonempty.

It remains to prove that the following equivalence: $\mathcal{T}$ has an infinite path $\pi$ with long-run frequency $lrf_{\mathcal{A}}(\pi)$ at least $\theta$ if and only if the intersection language of the $D_i$’s is nonempty.

Let us recall that, formally, the relation of $\mathcal{T}$ consists of the following transitions:

- $s_{i,1} \rightarrow s_{i,2} \rightarrow \ldots \rightarrow s_{i,k-1}$ for $i = 1, \ldots, k$
- $s_{i,1} \rightarrow s_{i+1,1}$ for $i = 1, \ldots, k-1$
- $s_{i,k-1} \rightarrow u_{\sigma} \rightarrow t_i$ for $\sigma \in \Sigma$
- $t_1 \rightarrow t_2 \rightarrow \ldots \rightarrow t_{k-1}$
- $t_{k-1} \rightarrow s_{1,1}$ and $t_{k-1} \rightarrow u_{\sigma}$ for $\sigma \in \Sigma$

Suppose first that there is some word $w \in \Sigma^*$ accepted by each of the DFSs $D_1, \ldots, D_k$. As stated before, we can safely assume that $|w| \leq \ell$. $\mathcal{T}$ has a cycle $\xi$ generating the word $\nu = s_1^{k-1}s_2^{k-1} \ldots s_k^{k-1}w$. We then have:

$$|\nu| = k(k-1) + k|w|$$

The word $\nu$ contains exactly $k(k-1) + (k-1)|w|$ positions from which a word accepted by $\mathcal{A}$ starts. This follows from the following two observations:

- The suffixes $s_1^{k-1} \ldots s_k^{k-1}w$ of $\nu$ are accepted by $\mathcal{B}_1$, and therefore by $\mathcal{A}$ (for $j = 1, \ldots, k-1$).
- $\nu$ contains exactly $(k-1) \cdot |w|$ positions from which a subword contained in $#^+$ starts.

Thus, the long-run frequency of the infinite path $\xi^{\omega}$ that repeats this cycle ad infinitum is:

$$lrf_{\mathcal{A}}(\xi^{\omega}) = f(|w|)$$

where $f: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is the following function:

$$f(x) = \frac{k(k-1) + (k-1)x}{k(k-1) + kx}$$

Function $f$ is monotonically decreasing. Therefore, $f(|w|) \geq f(\ell)$ as $f(\ell) = \theta$, we conclude that $\mathcal{T}$ has an infinite path with long-run frequency at least $\theta$.

We assume now that $\mathcal{T}$ has an infinite path $\pi$ with long-run frequency at least $\theta$. We first observe that $\pi$ must visit $s_{1,1}$ infinitely often as otherwise $\mathcal{A}$ would have an infinite suffix consisting of $t$- and $u$-states, in which case the long-run frequency would be smaller or equal than $(k-1)/k$, and therefore strictly smaller than $\theta$. Suppose by contradiction $\mathcal{L}(\mathcal{D}_1) \cap \ldots \mathcal{L}(\mathcal{D}_k)$ is empty. Then, the average weight obtained by each cycle $s_{1,1}s_{1,2} \ldots s_{1,k-1} \ldots s_{k,1}^k s_{2,2} \ldots s_{k,k-1}^k \omega s_{1,1}$ where $\omega$ consists of $t$- and $u$-states contained in $\pi$ in $\mathcal{T}$ is less or equal

$$\frac{(k-1)^2 + (k-1)y}{k(k-1) + ky} = \frac{k-1}{k}$$

where $y$ is the number of $u$-states in $\omega$, in which case the number of $t$-states in $\omega$ is $(k-1)y$. But then again, the long-run frequency of $\pi$ would be bounded by $(k-1)/k$, and therefore strictly smaller than $\theta$. Contradiction. We conclude that $\mathcal{L}(\mathcal{D}_1) \cap \ldots \mathcal{L}(\mathcal{D}_k)$ must be nonempty if $\mathcal{T}$ has an infinite path with long-run frequency at least $\theta$.

### Qualitative decision problems

**Lemma A.8.** Given a state $s$ of a KS $\mathcal{T}$ and NFA $\mathcal{A}$, the problem to check whether there is an infinite path $\pi$ of $\mathcal{T}$ starting in $s$ with

(a) $lrf_{\mathcal{A}}(\pi) > 0$ is in $P$,
(b) $lrf_{\mathcal{A}}(\pi) < 1$ is $\text{PSPACE}$-hard,
(c) $lrf_{\mathcal{A}}(\pi) = 0$ is $\text{NP}$-hard,
(d) $lrf_{\mathcal{A}}(\pi) = 1$ is $\text{NP}$-hard.

**Proof.** We first consider statement (a). As stated before, it suffices to consider the case where $\mathcal{T}$ is strongly connected, in which case the starting state $s$ is irrelevant. We build the synchronous product $\mathcal{T} \otimes \mathcal{A}$ and treat it as an NFA where $S \times Q_0$ is the set of initial states and $S \times F$ the set of final states. We now show: $\mathcal{T}$ has an infinite path $\pi$ with $lrf_{\mathcal{A}}(\pi) > 0$ if and only if the language of $\mathcal{T} \otimes \mathcal{A}$ is nonempty.

- For the implication “$\Rightarrow$” we pick a finite word $w$ over $\Sigma^+$ that is accepted by $\mathcal{T} \otimes \mathcal{A}$. As $Q_0 \cap F = \emptyset$, $w$ is non-empty. Pick an accepting run $(s_0, q_0)(s_1, q_1) \ldots (s_n, q_n)$ for $w$ in $\mathcal{T} \otimes \mathcal{A}$. We then pick a finite path $t_0, t_1, \ldots, t_m$ from $s_n = t_0 = s_0 = t_m$, and regard the infinite path $\pi$ arising by the infinite repetition of the cycle $s_0 s_1 \ldots s_n t_1 \ldots t_m$. Obviously, we then have $lrf_{\mathcal{A}}(\pi) > 0$.

1Each rational function $h(x) = (a + cx)/(b + dx)$ with $c b < a d$ is decreasing. This is a consequence of the fact that the first derivative is strictly negative. Note that $h'(x) = (c b - a d)/(b + dx)^2$, which is strictly negative if $c b < a d$. 

---

Fig. 4. The Kripke structure $\mathcal{T}$ in the reduction of Theorem III.7.
• For the implication \( \implies \), we suppose that we are given an infinite path \( \pi \) with \( \text{lref}_A(\pi) > 0 \). Then, there is a pair \((i,j)\) of integers with \( i < j \) such that the word induced by the path fragment \( \pi_{[i,j]} \) is accepted by \( A \). This path fragment can be lifted to an accepting run in \( T \otimes A \). Thus, the language of \( T \otimes A \) is nonempty.

We prove statement (b) via a polynomial reduction from the universality problem of finite automata, which is known to be \( \text{PSPACE} \)-complete [33], to the problem that takes as input a KS \( T \) and an NFA \( A \) and asks whether \( \text{lref}_A(\pi) = 1 \) for all infinite paths \( \pi \) in \( T \). Because \( \text{PSPACE} \) is closed under complement, this implies that checking the existence of an infinite path \( \pi \) such that \( \text{lref}_A(\pi) < 1 \) is \( \text{PSPACE} \)-hard. For the reduction from the universality problem for NFA we may restrict to NFA over the alphabet \( \Sigma = 2^A \) for some fixed set \( A \) of atomic propositions.

Given an NFA \( A = (Q, \Sigma, \delta, Q_0, F) \), we construct a new NFA \( B \) over the alphabet \( \Sigma' = \Sigma \cup \{\#\} \) for a new symbol \( \{\#\} \) as follows. The states of \( B \) are \( Q \) together with two new states \( s_{\text{out}} \) and \( \text{acc} \). The transitions in \( B \) are given by \( \delta \) and a new \#-transition from \( s_{\text{out}} \) to each state \( q \in Q_0 \), a new \#-transition from each state \( f \in F \) to \( \text{acc} \), and a new transition from \( s_{\text{out}} \) to \( \text{acc} \) for each symbol \( \sigma \in \Sigma \). The only accepting state in \( B \) is \( \text{acc} \) and the only initial state is \( s_{\text{out}} \). So, we get that \( L(B) = \Sigma \cup \{\#w\# \mid w \in L(A)\} \).

We define a KS \( T \) with state space \( S = \{s_\sigma : \sigma \in \Sigma'\} \), the obvious labeling function \( L(s_\sigma) = \sigma \), and with transitions between all pairs of states. Thus, all words on \( \Sigma' \) can be generated by \( T \).

Assume \( A \) is universal. Consider some infinite path \( \pi \) of \( T \). If \( \pi \) contains only finitely many \#s then \( \text{lref}_B(\pi) = 1 \), as all suffixes not starting with \# have a prefix accepted by \( B \) in one step. If \( \pi \) contains infinitely many \#s, then \( \text{lref}_B(\pi) = 1 \) as all suffixes not starting with \# have a prefix of length \( 1 \) which is accepted by \( B \) and all suffixes starting with \# have a prefix of the form \( \#w\# \) for some word \( w \). As \( A \) is universal, \( w \in L(A) \) and hence \( \#w\# \in L(B) \).

Conversely, assume that there is a word \( w \notin L(A) \). Then, the path \( \pi = (\#w)^\omega \) in \( T \) satisfies \( \text{lref}_B(\pi) = \frac{|w|}{|w|+1} < 1 \). As \( \pi \) starts with \#, the only prefix that could be accepted by \( B \) is \( \#w\# \). But this word is rejected by construction. So, every \( |w|+1 \)th suffix of \( \pi \) has no prefix accepted by \( B \).

Hence, we showed that there is a path \( \pi \) in \( T \) with \( \text{lref}_B(\pi) < 1 \) if and only if \( A \) is non-universal.

For the proof of statement (c), we describe a polynomial reduction from 3SAT. Let \( \psi = c_1 \lor \ldots \lor c_m \) be a 3CNF formula with \( m \) clauses using \( n \) Boolean variables, say \( x_1, \ldots, x_n \). Let \( T \) be a KS containing one state per clause \( c_i \), and for each \( c_i \), a fresh state per literal of \( c_i \). The initial state is \( c_i \). At state \( c_i \), there are three successors which are the literals of \( c_i \), which all go directly to \( c_{i+1} \). We create an additional state \( c_{m+1} = \bot \) whose only outgoing transition goes to \( c_1 \). The automaton \( A \) reads literals and guesses any contradiction in the input. It accepts iff for some \( i \), the word contains both \( x_i \) and \( \neg x_i \). Then, if \( \psi \) is satisfiable, there exists an infinite path \( \pi \) in \( T \) such that \( \text{lref}_A(\pi) = 0 \). If \( \psi \) is not satisfiable, for all \( \pi \) in \( T \), at each cycle, some \( x_i \) and \( \neg x_i \) must be read. Thus, \( \text{lref}_A(\pi) \geq \frac{1}{n+m+1} \).

The proof of statement (d) is also a reduction from 3SAT and uses the same KS \( T \) as in the previous construction. Consider 3CNF formula \( \psi = c_1 \lor \ldots \lor c_m \) over variables \( x_1, \ldots, x_n \), and let us write \( c_i = l_{1i} \lor l_{2i} \lor l_{3i} \) where \( l_{1i}, l_{2i}, l_{3i} \) are literals. Let \( L \) denote the set of all literals, and \( C = (c_1, \ldots, c_m) \). We define \( A \) as follows. From the initial state, one goes to an accepting sink state by reading any letter of \( L \cup \{\bot\} \). By reading \( c_i \in C \), one goes to \( s_{c_i} \) from which the automaton accepts when \( \bot \) is read iff one of the literals of \( c_i \) was seen in the mean time, but not its negation. That is, if none of the literals of \( c_i \) are seen, or some literal \( l_{1i} \) and its negation are both seen, then the automaton ends in a rejecting sink state. This part of the automaton can be constructed using 3 extra bits in order to store which literals have been seen.

If \( \psi \) is satisfiable, then there is a word \( w = c_1 l_{1i} l_{2i} \ldots c_m l_{mi} \bot \) which can be written as \( w = l_{1k} \lor \ldots \lor l_{mk} \bot \) for some \( k \). Assume \( k \neq k' \). Consider the word \( \pi = w^{\#\#} \). The property \( A \) is satisfied from all positions. This is trivial from positions containing \( X \cup \{\bot\} \). At any other position containing \( c_i \), the letter is followed by an actual literal of \( c_i \), and the rest of the word does not contain its negation. Thus, \( \text{lref}_A(\pi) = 1 \).

Assume now that \( \psi \) is not satisfiable. Consider any word \( \pi \) of \( T \) which can be written as \( \pi = w_1 \bot w_2 \bot \ldots \) where each \( w_i \) has the form \( c_1 l_{1i} l_{2i} \ldots l_{mi} \bot \) with \( l_{1k} \in c_k \) for each \( k \). Since \( \psi \) is not satisfiable, each \( w_i \) must contain \( l_1 \) and \( l_{-1} \) with \( l_1 = l_{-1} \). That is, at least once every \( m+n+1 \) positions, the property is violated. Thus, \( \text{lref}_A(\pi) < 1 \).

\[ \square \]

C. Proofs for the probabilistic case

Additional notations: End components (ECs) have been introduced in Section II as sets of state-action pairs where the induced graph is strongly connected. Occasionally, we shall use a representation of an end component \( E \) as a pair \( (E, \mathcal{A}) \) where \( E \) is a set of states and \( \mathcal{A} : E \rightarrow \mathcal{Act} \), namely \( E = \{s \in S : \exists \alpha \in \mathcal{Act}(s, \alpha) \in E \} \) and \( \mathcal{A}(s) = \{\alpha \in \mathcal{Act}(s, \alpha) \in E \} \). With this representation, mind and identifying \( E \) with \( E \), we sometimes use notations like \( s \in E \) or \( T \cap E \) for \( s \in S \) and \( T \subseteq S \).
If $\mathcal{S}$ is a scheduler and $\pi = s_0a_0s_1a_1 \ldots$ a path then $\pi$ is said to be a $\mathcal{S}$-path if $\mathcal{S}(s_0a_0 \ldots a_{n-1} s_n)(s_n) > 0$ for all $n \geq 0$. Many proofs will rely on de Alfaro’s observation [19] stating that for each scheduler of a (finite-state) MDP, the limit of almost all infinite $\mathcal{S}$-paths constitutes an end component. Here, the limit $\text{Lim}(\pi)$ of an infinite path $\pi$ denotes the set of state-action pairs that occur infinitely often in $\pi$.

**Efficiently solvable instances of the lrp-problem**

**Theorem A.9** (see Theorem IV.3). The values $\text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi)$ and $\text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi)$ are computable in polynomial-time if $\varphi$ is a condition of one of the following types:

- reachability $\Diamond b$,
- invariance $\Box b$,
- generalized Rabin $\bigwedge_{i=1}^{n} (\Diamond \Box b_{i,j} \lor \Diamond \Box a_{i,j})$
- or Streett $\bigwedge_{i=1}^{n} (\Box \Diamond a_{i,j} \lor \Diamond \Box b_{i,j})$.

In all these cases, optimal FMD-scheduler exist. Moreover, optimal MD-schedulers exist for reachability, invariances, Büchi and co-Büchi conditions.

**Proof.** We provide the argument for $\text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi)$. The argument for $\text{LP}_{\mathcal{M},s}^{\text{min}}(\varphi)$ is analogous and omitted here. As stated above, we may assume that $\mathcal{M}$ is strongly connected.

It is well-known [2], [12], [19] that for all properties listed in Theorem IV.3 there is an FMD-scheduler $\mathcal{S}$ that maximizes the probability for $\varphi$ from every visited state in the following sense: $\text{LP}_{\mathcal{M},s,t}^{\mathcal{S}}(\varphi) = \text{LP}_{\mathcal{M},s,t}^{\mathcal{S},\pi}(\varphi)$ for each infinite $\mathcal{S}$-path $\pi$ and each position $i \in \mathbb{N}$. For reachability, invariances, Büchi and co-Büchi conditions, we may even suppose that $\mathcal{S}$ is an MD-scheduler with a single BSCC $\mathcal{B}$.

If $\varphi$ is a reachability, generalized Rabin or Streett condition then $\text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi) = \text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi)$ for all states $s,t \in \mathcal{M}$. Moreover, this value is either 0 or 1. But then $\mathcal{S}$ obviously achieves the maximal long-run probability from every state.

The states in $\mathcal{M}$ can have different maximal probabilities for invariances $\varphi = \Box b$. However, for invariances we either have $\text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi) = 0$, in which case $\text{LP}_{\mathcal{M},s}^{\text{max}}(\varphi) = 0$ for all states $s$, or the unique BSCC $\mathcal{B}$ of $\mathcal{S}$ consists of b-states. In the latter case, $\text{LP}_{\mathcal{M},s}^{\text{max}}(\Box b) = \text{LP}_{\mathcal{M},s}^{\text{max}}(\Box b) = 1$ for all states $s$ in $\mathcal{B}$. Let now $\mathcal{S}$ be the following MD-scheduler:

- For the state inside $\mathcal{B}$, $\mathcal{S}$ mimics an MD-scheduler maximizing the probability to reach $\mathcal{B}$ (which is 1 as $\mathcal{M}$ is strongly connected).

We then have $\text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\Box b) = 1$ for all states $s$ in $\mathcal{M}$, which is obviously maximal.

**Qualitative lrp-problems**

Recall that the task of the qualitative lrp-problems is to decide the existence of a scheduler $\mathcal{S}$ such that $\text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi)$ is positive, equals 1, is strictly less than 1 or equals 0.

**Lemma A.10.** The four qualitative lrp-problems for MDPs and until properties are decidable in polynomial time. Moreover, if $\mathcal{M}$ is strongly connected and $s$ a state of $\mathcal{M}$ then:

\[
\text{LP}_{\mathcal{M}}(a \cup b) = 1 \iff \exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) = 1
\]

\[
\text{LP}_{\mathcal{M}}^{\text{min}}(a \cup b) = 0 \iff \exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) = 0
\]

**Proof.** Let $\varphi = a \cup b$ and $\mathcal{M}$ be a strongly connected MDP. Polynomial-time decidability of the four qualitative lrp-problems is a direct consequence of the following observations:

(a) $\exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) > 0$ iff $B \neq \emptyset$

(b) $\exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) = 1$ iff $\mathcal{M}$ has a $(A \cup B)$-EC with $E \cap B \neq \emptyset$ or $\mathcal{M}$ has an A-EC $E$ with $\text{Pr}_{\mathcal{E}}^{\text{max}}(\varphi) = 1$ (or both).

(c) $\exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) < 1$ iff $\mathcal{M}$ has an A-EC or $C \neq \emptyset$

(d) $\exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) = 0$ iff $\mathcal{M}$ has a $(A \cup C)$-EC

Statement (a) is obvious. For the proof of statement (b), we show the equivalence of the following three statements:

(i) $\text{LP}_{\mathcal{M}}^{\text{max}}(\varphi) = 1$

(ii) $\exists S. \text{LP}_{\mathcal{M},s}^{\mathcal{S}}(\varphi) = 1$

(iii) $\mathcal{M}$ has a $(A \cup B)$-EC $E$ with $E \cap B \neq \emptyset$ or $\mathcal{M}$ has an A-EC $E$ with $\text{Pr}_{\mathcal{E}}^{\text{max}}(\varphi) = 1$ (or both).

"(iii) $\iff$ (ii)". The claim is obvious if $\mathcal{M}$ has a $(A \cup B)$-EC $E$ with $E \cap B \neq \emptyset$. Suppose now that $\mathcal{M}$ has an A-EC $E$ with $\text{Pr}_{\mathcal{E}}^{\text{max}}(\varphi) = 1$. An infinite-memory scheduler with long-run probability 1 can be obtained by using the A-EC $E$ to have longer and longer portions of A-states, before satisfying almost-surely $a \cup b$ (by following an MD-scheduler maximizing the probability for $a \cup b$ and returning to $E$ from the reached B-state (by following an MD-scheduler maximizing the probability for reaching $E$ from every state in $\mathcal{M}$).

"(i) $\iff$ (iii)". By contradiction. Suppose that none of the two alternative condition holds. Then, C-states are seen "often" on average in the following sense: if $n$ denotes the number of states then $\text{LP}_{\mathcal{M},t}^{\text{max}}(\varphi \land \neg C) > 0$ for every state $s$. As $\text{LP}_{\mathcal{M},t}^{\text{max}}(a \cup b) = 0$ for each C-state $t$, this yields

\[
\text{LP}_{\mathcal{M},s}^{\mathcal{S}}(a \cup b) \leq \frac{n-1}{n} < 1
\]
for each scheduler $\mathcal{S}$. Hence, $L_{\text{max}}^{\text{MD}}(a U b) \leq (n-1)/n < 1$.

The implication “(ii) $\implies$ (i)” is trivial.

The proofs for statements (c) and (d) and the equivalence of $L_{\text{max}}^{\text{MD}}(a U b) = 0$ and the existence of a scheduler $\mathcal{S}$ with $L_{\text{max}}^{\mathcal{S}}(a U b) = 0$ are similar and omitted here.

\[\Box\]

Duality of minimal and maximal long-run probabilities

The following lemma essentially shows that minimizing the long-run probability for $a U b$ is dual to the task to maximize the long-run probability for $A U C$.

An exception is the case where $M$ has A-ECs, in which case the minimal lrp-problem can be answered directly.

**Lemma A.11 (Min-lrp via max-lrp).** Suppose $M$ is a strongly connected and let $A, B, C$ as above. If $M$ has an A-EC then $L_{\text{max}}^{\text{MD}}(a U b) = 0$. Otherwise, i.e., if $M$ has no A-EC, then $L_{\text{min}}^{\text{MD}}(a U b) = 1 - L_{\text{max}}^{\text{MD}}(A U C)$.

**Proof.** Suppose first that $M$ has an A-EC $E$. For each state $t$ in $E$, let $A(t) = \{s \in \text{Act}: (s, \alpha) \in E\}$. Consider an MD-scheduler $\mathcal{S}$ minimizing the probability to reach $E$ from every state $s$ outside $E$ and which selects only actions in $A(t)$ for every state $t$ in $E$. Then, $\Pr_{\mathcal{S},s}^{\text{MD}}(A U b) = 0$ (as $A$ and $B$ are disjoint). But then $L_{\text{min}}^{\text{MD}}(A U b) = L_{\text{max}}^{\text{MD}}(A U C) = 0$ for each state $s$. This yields $L_{\text{min}}^{\text{MD}}(a U b) = 1 - L_{\text{max}}^{\text{MD}}(A U C)$.

Suppose now that there are no A-ECs. Then, for each scheduler $\mathcal{S}$ and each state $s$:

$$\Pr_{\mathcal{S},s}^{\text{MD}}(a U b) = \Pr_{\mathcal{S},s}^{\text{MD}}(A U B) = 1 - \Pr_{\mathcal{S},s}^{\text{MD}}(A U C)$$

This yields $L_{\text{min}}^{\text{MD}}(a U b) = 1 - L_{\text{max}}^{\text{MD}}(A U C)$. \[\Box\]

Finite-memory schedulers for $a U b$

We now show that the maximal long-run probabilities for until properties can be approximated by FM-schedulers.

As before, we suppose that $M$ is a strongly connected MDP with state space $S$. Furthermore, we may safely assume that $C \not\subseteq S$ as otherwise all states either belong to $A$ or $B$, in which case either $L_{\text{max}}^{\text{MD}}(a U b) = 0$ if $B = \emptyset$ or, if $B \neq \emptyset$, then $L_{\text{max}}^{\text{MD}}(a U b) = L_{\text{max}}^{\text{MD}}(a U b) = 1$ for any MD-scheduler $\mathcal{S}$ that maximizes the probability for reaching $B$.

**Lemma A.12 (see Lemma IV.6).** For each scheduler $\mathcal{T}$ for $M$, each $\epsilon > 0$ and each state $s$ of $M$, there is an FM-scheduler $\mathcal{S}$ for $M$ such that:

$$L_{\text{max}}^{\mathcal{S}}(a U b) \geq L_{\text{max}}^{\mathcal{T}}(a U b) - \epsilon$$

**Proof.** If $\mathcal{S}$ is a scheduler then briefly we write $p_{\mathcal{S}}^{\mathcal{T}}$ instead of $\Pr_{\mathcal{S},s}^{\mathcal{T}}(A U B)$.

By Fatou’s lemma, we have:

$$L_{\text{max}}^{\mathcal{S}}(a U b) = \liminf_{n \to \infty} \frac{1}{n+1} \sum_{i=0}^{n} \sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{n}$$

$$\leq \liminf_{n \to \infty} \sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{n}$$

So, there exists $k_0 \in \mathbb{N}$ such that for all $k \geq k_0$:

$$\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1} \leq \sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k}$$

Let $\mathcal{U}_k$ be the following FM-scheduler with two modes. If the current state is in $A$, it starts in the first mode, in which it behaves like an MD-scheduler maximizing the probability of $a U b$. As soon as a state in $B \cup C$ has been reached, scheduler $\mathcal{U}_k$ operates in the second mode, in which it memorylessly minimizes the expected number of steps until reaching $s$. Let $f_{\mathcal{U}_k}^{\mathcal{T}} = \sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \min_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} (\text{steps until } s)$ denote the expected number of steps this scheduler $\mathcal{U}_k$ needs to reach $s$ in the second mode starting from state $t$. Then define $f_s = \max_{t \in S} f_{\mathcal{U}_k}^{\mathcal{T}}$ and $f = \max_{s \in S} f_s$.

We now construct an FM-scheduler $\mathcal{S}$ satisfying the claim of the lemma. First, choose a natural number $k$ with $k \geq 0$ and $k+1 > 2f_s$. The behavior of scheduler $\mathcal{S}$ is as follows. In its first mode, it starts in $s$ and behaves like $\mathcal{T}$ in the first $k$ steps. Then, it switches to the second mode and behaves like $\mathcal{U}_k$ until it reaches $s$ (in the second mode of $\mathcal{U}_k$). Afterwards, it switches back to the first mode.

As $\mathcal{U}_k$ maximizes the probability of $a U b$ whenever it starts in a state in $A$, we obtain:

$$P_{\mathcal{S}}^{\mathcal{T}}(a U b) \leq \frac{1}{k+1} \sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}$$

for all paths $\pi$. Furthermore, the expected number of steps which $\mathcal{S}$ takes to follow $\mathcal{T}$ for $k+1$ steps and to return to $s$ via $\mathcal{U}_k$ is at most $k+1+f_s$.

Expressing the long-run probability of $\mathcal{S}$ as a quotient, we obtain:

$$L_{\text{max}}^{\mathcal{S}}(a U b) = \frac{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \tau_{\pi}^{(a U b)}}{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}}$$

$$\geq \frac{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \tau_{\pi}^{(a U b)}}{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}}$$

$$\geq \frac{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \tau_{\pi}^{(a U b)}}{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}}$$

$$\geq \frac{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \tau_{\pi}^{(a U b)}}{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}}$$

$$\geq \frac{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \tau_{\pi}^{(a U b)}}{\sum_{\pi \in \mathcal{P}_{\mathcal{S}}^{\mathcal{T}}(a U b)} \frac{\tau_{\pi}^{(a U b)}}{k+1}}$$
by the choice of $k$. Using the fact that $\mathbb{L}^{\pi^T}_{M,s}(a \cup b)$ is bounded by 1 we obtain:

$$\mathbb{L}^{\pi^T}_{M,s}(a \cup b) \geq (\mathbb{L}^{\pi^T}_{M,s}(a \cup b) - \epsilon/2) \cdot (1 - \epsilon/2) \geq \mathbb{L}^{\pi^T}_{M,s}(a \cup b) - \epsilon$$

This completes the proof of Lemma A.12 and yields the statement of Lemma IV.6.

\[\square\]

**Saturation point**

We present the proof of Lemma IV.7. Let us first recall the relevant notations. The value $K$ has been defined in Section IV of the main paper as $\max|N,[e/\delta]|$ where $N = |A|$ and where $e$ and $\delta$ are as follows.

The value $\delta$ has been defined $\delta = \min_{s \in A} \delta_s$ where $\delta_s$ denotes the minimal difference between $p^\text{max}_s = \mathbb{P}^\text{max}_{M,s}(a \cup b)$ and the values $p_s,\alpha = \sum_{t \in S} \mathbb{P}(s, \alpha, t) \cdot p^\text{max}_t$ for $\alpha \in \text{Act}(s) \setminus \text{Act}^{\max}(s)$, and $\delta_s = \infty$ if $\text{Act}(s) = \text{Act}^{\max}(s)$. In the special case where $\delta_s = \infty$ for all states $s \in A$, $\delta$ is defined to be 1.

For the definition of $e$ we considered the unfolded MDP $M_N$ (where again $N = |A|$) and defined $e$ as the maximum of the values

$$e_{t,s} = E^{\text{min}}_{M_N,1,t,s} \left( \text{"steps until } s_N \text{"} \right)$$

Here, $s_N = s$ for $s \in B \cup C$ and $s_N$ is any copy $(s, i)$ of state $s$ in $M_N$ that is reachable from some $(B \cup C)$-state in $M_N$. As $N = |A|$, with $M$ also $M_N$ is strongly connected. This ensures the well-definedness (finiteness) of the values $e_{t,s}$ and the existence of an MD-scheduler $\mathfrak{R}_{N,s}$ for $M_N$ that minimizes the expected number of steps to $s_N$ from every state $t_N$. This MD-scheduler $\mathfrak{R}_{N,s}$ for $M_N$ can be viewed as an FMD-scheduler for $M$. We then have:

$$e_{t,s} \geq E^{\text{min}}_{M_N,1,t,s} \left( \text{"steps until } s \text{"} \right)$$

for all states $s, t \in S$. Moreover, $\mathfrak{R}_{N,s}$ enjoys the property that $\mathbb{P}^{\mathfrak{R}_{N,t} \in \omega}_{M,t}(a \cup b) = p^\text{max}_t$ for each $\mathfrak{R}_{N,s}$-path $\omega$ that has a suffix consisting of $N$ or more $A$-states where $t = \text{last}(\omega)$. In particular, $\mathfrak{R}_{N,s}$ only schedules actions in $\text{Act}^{\max}(\cdot)$ when having generated a path that ends with a suffix consisting of $N$ or more $A$-states.

**Lemma A.13 (see Lemma IV.7).** Suppose $M$ has no $(A \cup B)$-EC containing at least one $B$-state. Then, for each $FM$-scheduler $\Sigma$, there is a scheduler $\Theta \in \text{FM}(K)$ with $\mathbb{L}^{\text{max}}_{M,N}(a \cup b) \geq \max_{s \in S} \mathbb{L}^{\pi^T}_{M,s}(a \cup b)$.

**Proof.** Let $\Sigma$ be an FMD-scheduler for $M$ with modes (memory cells) in the finite set $X$. Let $\mathcal{C}^T$ denote the Markov chain induced by $\Sigma$. We can think of the states in $\mathcal{C}^T$ as pairs $(s, x)$ consisting of a state $s$ in $M$ and a mode $x \in X$. We may assume w.l.o.g. that $\mathcal{C}^T$ has a single BSCC, say $\mathcal{B}^T$. This yields that all states of $\mathcal{C}^T$ have the same long-run probability for $a \cup b$. Let us simply write $\mathbb{L}^{\pi^T}_{M}(a \cup b)$ for this value.

Given a state $s = (s, x)$ in $\mathcal{C}^T$, we say $s$ is an $A$-state if $s \in A$. The notations $B$-state and $C$-state have the analogous meaning. We suppose that all $A$-states of $\mathcal{C}^T$ are labeled with $a$, while the $B$-states are labeled with $b$.

If $\mathcal{B}^T$ consists of $(A \cup C)$-states then $\mathbb{L}^{\pi^T}_{M}(a \cup b) = 0$ and the claim is trivial as we can deal with any $\text{FM}(K)$-scheduler.

Suppose now that $\mathcal{B}^T$ contains at least one $B$-state. Then, almost all $\Sigma$-paths visit infinitely often some $B$-state and infinitely often some $C$-state.

We now explain how to modify $\Sigma$’s decision for generating a scheduler in $\text{FM}(K)$ with the desired property. Our procedure works by induction on the number of $(B \cup C)$-states $s = (s, x)$ in $\mathcal{B}^T$ where

$$\mathbb{P}_{\mathcal{B}^T,d}(\bigcirc(A \cup \Gamma K D^T)) > 0$$

Here, $D^T$ denotes the set of of $A$-states $t = (t, y)$ in the BSCC $\mathcal{B}^T$ where $\Sigma(t)(\alpha) > 0$ for some action $\alpha \notin \text{Act}^{\max}(t)$.

If $k^T = 0$ then for every path $\omega = s_0, a_0 \ldots s_{n-1}, s_n$ in $\mathcal{B}^T$ of length $n \geq K$ where all but the first state are $A$-states, the states $s_0, s_1, \ldots, s_n$ do not belong to $D^T$. That is, $\Sigma$ schedules only actions in $\text{Act}^{\max}$ for these states. But then the probability for a $a \cup b$ in $\mathcal{B}^T$ from each of the states $s_i = (s_i, x_i)$ with $i \geq K$ equals $p^\text{max}_{s_i}$. This implies that if $k^T = 0$ then $\Sigma$ is an $\text{FM}(K)$-scheduler and we can deal with $\Theta = \Sigma$.

Suppose now that $k^T > 1$. We show how to transform $\Sigma$ into a new $\text{FM}$-scheduler $\Theta$ with a single BSCC such that $\mathbb{L}^{\Theta}_{M}(a \cup b) \geq \mathbb{L}^{\pi^T}_{M}(a \cup b)$ and $k^\Theta < k^T$.

Given states $s = (s, x)$ and $t = (t, y)$ in $\mathcal{B}^T$, where $s$ is a $(B \cup C)$-state and $t$ an $A$-state, let $\Gamma_{s,t}$ denote the set of of finite $\Sigma$-paths $\omega = s_0, a_0 \ldots a_{n-1}, s_n$ with

- $n \geq K$,
- $s_0 = s$, $s_n = t$,
- $s_1, \ldots, s_n$ are $A$-states, and
- $\Sigma(s_n)(\alpha) > 0$ for some action $\alpha \notin \text{Act}^{\max}(t)$.

Let $\Pi_{s,t}$ denote the set of paths $\omega \in \Gamma_{s,t}$ such that no proper prefix of $\omega$ belongs to $\Gamma_{s,t}$, and let $\Pi_{s}$ denote the union of the sets $\Pi_{s,t}$.

As $k^T$ is positive, we can pick some $(B \cup C)$-state $s = (s, x)$ in $\mathcal{B}^T$ where $\Pi_{s}$ is nonempty.

The definition of $\text{FM}$-scheduler $\Theta$ is as follows. Scheduler $\Theta$ operates in two phases. Its first phase starts in

\[\text{Proof.} \]
\( s = (s, x) \) and uses additional memory cells to keep track of the number of consecutive A-states that have been traversed since the last visit of \( s \). More precisely, if \( \mathcal{G} \) has generated the path \( s_0 s_1 \ldots s_i \) where \( i \leq K \), \( s_0 = s \) and \( s_1, \ldots, s_i \) are A-states then the memory cells encode the counter value \( i \). As long as the counter value is smaller than \( K \) or if a \( B \cup C \)-state has been reached along a path where the counter value is always smaller than \( K \), scheduler \( \mathcal{G} \) just behaves like \( \mathcal{T} \). As soon as the counter value equals \( K \), scheduler \( \mathcal{G} \) switches to the second phase and behaves as scheduler \( \mathcal{R}_{N,s} \). More precisely, if \( \mathcal{G} \)'s current state \( t \) in \( \mathcal{M} \) belongs to \( A \) then \( \mathcal{G} \) mimics the behavior of \( \mathcal{R}_{N,s} \) from state \( (t, \top) \). Thus, by following \( \mathcal{R}_{N,s} \)'s decisions, \( \mathcal{G} \) will only choose actions in \( \text{Act}^\text{max} \) until \( A \) is left.

As soon as state \( s \) is reached in \( \mathcal{G} \)'s second phase (this will happen with probability \( 1 \) as \( \mathcal{R}_{N,s} \) minimizes the expected number of steps to \( s = s_N \) from every state in the strongly connected MDP \( \mathcal{M}_N \) ). \( \mathcal{G} \) switches back to the first phase and restarts to mimic from state \( s \) in mode \( x \), i.e., from state \( s \) in \( \mathcal{B}^\top \). For all states that are not reachable from \( s \) in this way, \( \mathcal{G} \) behaves as \( \mathcal{T} \).

As \( \mathcal{T} \) has a single BSCC, so does \( \mathcal{G} \), although the BSCC \( \mathcal{B}^\mathcal{G} \) induced by \( \mathcal{G} \) can be different from \( \mathcal{B}^\top \).

As \( s \) belongs to both \( \mathcal{B}^\top \) and \( \mathcal{B}^\mathcal{G} \), \( s \) is visited infinitely often almost surely with finite expected return time under both schedulers \( \mathcal{G} \) and \( \mathcal{T} \).

Let us first observe that we indeed have \( k^\mathcal{G} < k^\top \).

This is thanks to the fact that (1) \( \mathcal{R}_{N,s} \) maximizes the probability for \( a \cup b \) whenever \( N \) or more consecutive A-states have been visited, (2) \( K \geq N \) and (3) the reference state \( s \) is not an A-state. Thus, for each \((B \cup C)\)-state \( u \) visited by \( \mathcal{R}_{N,s} \) in the return (second) phase of \( \mathcal{G} \) we have:

\[
Pr_{B^\mathcal{G},(u)}(\bigcap(\mathcal{A}U^\geq K D^\mathcal{G})) = 0
\]

Hence, whenever \( u \) is \((B \cup C)\)-state in \( \mathcal{B}^\mathcal{G} \) where \( Pr_{B^\mathcal{G},(u)}(\bigcap(\mathcal{A}U^\geq K D^\mathcal{G})) \) is positive then

- \( u \neq s \),
- \( u \) also belongs to \( \mathcal{B}^\top \) and
- the \( \mathcal{G} \)-paths from \( u \) satisfying \( \bigcap(\mathcal{A}U^\geq K D^\mathcal{G}) \) are also \( \mathcal{T} \)-paths and satisfy \( \bigcap(\mathcal{A}U^\geq K D^\top) \).

The last item yields \( Pr_{B^\mathcal{T},(u)}(\bigcap(\mathcal{A}U^\geq K D^\top)) > 0 \). This completes the proof that \( k^\mathcal{G} \) is smaller than \( k^\top \).

We now show that \( \mathbb{L}Pr_{\mathcal{M}}^\mathcal{G}(a \cup b) > \mathbb{L}Pr_{\mathcal{M}}^\top(a \cup b) \). To simplify the calculations, we present the proof for the case where \( \Pi_\alpha \) is a singleton, say \( \Pi_\alpha = \{\alpha\} \).

Furthermore, let \( n \) be the length of \( \omega \) (then \( n \geq K \)) and \( t = (t, y) = \text{last}(\omega) \). Again, to simplify the calculations, let us suppose that there is a single action \( \alpha \in \text{Act}(t) \setminus \text{Act}^\text{max}(t) \) that \( \mathcal{T} \) schedules for \( t \) with positive probability \( p \). So,

\[
p = \mathbb{T}(t)(\alpha) > 0
\]

(Note that \( p = 1 \) if \( \mathcal{T} \) is a deterministic FM-scheduler.)

The long run probabilities of the two schedulers \( \mathcal{G} \) and \( \mathcal{T} \) can be expressed as follows.

Given a state \( u \) in \( \mathcal{B}^\top \), let \( ens_{u,s}^\mathcal{G} \) be the expected number of steps from \( u \) to \( s \) under \( \mathcal{T} \) (via paths from \( u \) to \( s \) of length at least \( 1 \) where all intermediate states are different from \( s \)). Let \( eap_{u,s}^\mathcal{G} \) denote the expected accumulated probability for \( a \cup b \) that \( \mathcal{T} \) accumulates during this period. So, \( ens_{u,s}^\mathcal{G} \) can be understood as the expected return time from and to \( s \) under \( \mathcal{T} \). Then:

\[
\mathbb{L}Pr_{\mathcal{M}}^\mathcal{G}(a \cup b) = \frac{eap_{s,0}^\mathcal{G}}{ens_{s,0}^\mathcal{G}} (\dagger)
\]

For the scheduler \( \mathcal{G} \) we express \( \mathbb{L}Pr_{\mathcal{M}}^\mathcal{G}(a \cup b) \) as the fraction of the expected accumulated probability along return paths and the expected return time from \( s \) to \( s \) as well:

\[
\mathbb{L}Pr_{\mathcal{M}}^\mathcal{G}(a \cup b) = \frac{eap_{s,0}^\mathcal{G}}{ens_{s,0}^\mathcal{G}}
\]

To provide an upper bound for \( ens_{s,0}^\mathcal{G} \) and a lower bound for \( eap_{s,0}^\mathcal{G} \), we need several auxiliary notations.

Recall that \( e_{t,s} \) is the expected number of steps that \( \mathcal{R}_{N,s} \) needs from \( t_N \) to \( s_N \). Hence, \( e_{t,s} \) is an upper bound for the expected number of steps \( ens_{t,s}^\mathcal{G} \) that \( \mathcal{G} \) needs from state \( t = \text{last}(\omega) \) to the reference state \( s \).

The value \( e \) has been defined as the maximum of the values \( e_{t,s} \). Hence, we obtain:

\[
e \geq e_{t,s} \geq ens_{t,s}^\mathcal{G}
\]

Let \( ens_{t,s}^\mathcal{G} \) denote the the expected number of steps that \( \mathcal{T} \) needs from \( t \) to \( s \), under the assumption that action \( \alpha \) is scheduled in \( t \) (which happens with probability \( p \)). So, if \( \{u_1, z_1\}, \ldots, \{u_{\ell}, z_{\ell}\} \) denote the \( \alpha \)-successors of \( t \) in \( \mathcal{B}^\top \) then:

\[
ens_{t,s}^\mathcal{G} = 1 + \sum_{i=1}^{\ell} P(t, \alpha, u_i) \cdot ens_{(u_i, z_i), s}^\mathcal{G}
\]

Similarly, we define:

\[
eap_{t,s}^\mathcal{G} = Pr_{\mathcal{M}}^\mathcal{G}(a \cup b) + \sum_{i=1}^{\ell} P(t, \alpha, u_i) \cdot eap_{(u_i, z_i), s}^\mathcal{G}
\]

For \( 0 \leq i \leq n \), let \( \varphi_{[0...i]} \) denote the probability under \( \mathcal{T} \) for generating the path fragment \( \omega_{[0...i]} \) from state \( s \) in mode \( x \). So, \( \varphi_{[0...n]} \) is the probability under \( \mathcal{T} \) for generating the full path \( \omega \) from \( s \).

---

\(^3\)At the end of the proof, we briefly explain how to treat the general case where \( \Pi_\alpha \) is a (prefix-free) countable set of paths, for which \( \mathcal{T} \) can schedule multiple actions not in \( \text{Act}^\text{max} \) with positive probability.
For the expected number of steps \( \text{ens}^{\mathcal{S}}_{s,s} \) that \( \mathcal{S} \) needs from \( s \) to \( s \) along paths of length at least 1, we get:
\[
\text{ens}^{\mathcal{S}}_{s,s} \leq \text{ens}^{\mathcal{S}}_{s,s} + \varphi_{[0..n]} \cdot p \cdot (e - \text{ens}^{\mathcal{S}}_{t,\alpha,s})
\]

Proof of (*) Let us first observe that \( \text{ens}^{\mathcal{S}}_{s,s} \) can be written as an infinite sum:
\[
\text{ens}^{\mathcal{S}}_{s,s} = \sum_{\zeta \in \Xi} \text{Pr}^{\mathcal{S}}(\zeta) \cdot |\zeta|
\]
where \( \Xi \) denotes the set of all finite paths \( \zeta \) that start in \( s \) and end in \( s \), and none of the intermediate states equals \( s \).

Just to simplify the notations, let us suppose that \( T \) all paths intermediate state equals \( s \) and vice versa, each \( T(t)(\mathcal{S}) \).

As \( \text{Pr}(\cdot) \) again is bounded by \( [0,1] \), and \( \mathrm{prf}(\cdot) \) denote the set of proper prefixes of the paths in \( \Xi \). Furthermore, we write \( \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b) \) for the probability for \( a U b \) in the Markov chain induced by \( \mathcal{S} \) from state \( \text{last}(\zeta) \).

Then, \( \text{ens}^{\mathcal{S}}_{s,s} \) can be written as an infinite sum:
\[
\text{ens}^{\mathcal{S}}_{s,s} = \sum_{\pi \in \Xi} \text{Pr}^{\mathcal{S}}(\pi) \cdot \sum_{\zeta \in \text{prf}(\pi)} \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b)
\]
\[
= \sum_{\pi \in \Xi} \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b) \cdot \sum_{\pi \in \Xi} \text{Pr}^{\mathcal{S}}(\pi)
\]
\[
= \sum_{\pi \in \Xi} \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b)
\]

We deal here with the proper prefixes of the paths in \( \Xi \) to avoid that the reference state \( s \), which is the starting and end state of the paths in \( \Xi \), is considered in the sums for the accumulated weight (probability values) of the paths \( \pi \in \Xi \).

The analogous formula holds for \( \Xi \). Let \( \Xi, \Xi^{\mathcal{S}} \) and \( \Xi^{\mathcal{S}} \) be defined as in the proof of (*). For the paths \( \zeta \) that are prefixes of some path in \( \Xi, \Xi^{\mathcal{S}} \), but not a prefix of \( \Xi \), we have \( \text{Pr}^{\mathcal{S}}(\zeta) = \text{Pr}(\zeta) \) and \( \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b) = \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b) \).

Thus, the difference between \( \text{ens}^{\mathcal{S}}_{s,s} \) and \( \text{ens}^{\mathcal{S}}_{s,s} \) is:
\[
\sum_{\pi \in \Xi^{\mathcal{S}} \text{prf}(\Xi^{\mathcal{S}})} \text{Pr}^{\mathcal{S}}(\zeta) \cdot \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b)
\]
\[
- \sum_{\pi \in \Xi^{\mathcal{S}} \text{prf}(\Xi^{\mathcal{S}})} \text{Pr}^{\mathcal{S}}(\zeta) \cdot \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b)
\]

The prefixes \( \varphi_{[0..n]} \) of \( \varphi \) are contained in both sets \( \text{prf}(\Xi^{\mathcal{S}}) \) and \( \text{prf}(\Xi^{\mathcal{S}}) \). These have the same probability under \( \mathcal{S} \) and \( \Xi \) (namely \( \varphi_{[0..n]} \)) and for \( t = 1, \ldots, n \):
\[
\text{Pr}^{\mathcal{S}}_{\varphi_{[0..n]}}(a U b) = \text{Pr}^{\mathcal{S}}_{\varphi_{[0..n]}}(a U b)
\]
\[

\text{Proof of (**). Let } \Xi \text{ be as in the proof of (*) and let } \mathrm{prf}(\cdot) \text{ denote the set of proper prefixes of the paths in } \Xi. \text{ Furthermore, we write } \text{Pr}^{\mathcal{S}}_{\text{last}(\zeta)}(a U b) \text{ for the probability for } a U b \text{ in the Markov chain induced by } \mathcal{S} \text{ from state } \text{last}(\zeta).
state, we get:
\[
\sum_{\zeta \in \text{pref}(\omega)} \Pr_S^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta) (a \cup b) - \sum_{\zeta \in \text{pref}(\omega)} \Pr_T^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta) (a \cup b) = \sum_{i=1}^n \theta_{[0 \cdots i]} \cdot \left( \Pr_S^\omega_{\alpha} (a \cup b) - \Pr_S^\omega_{\beta} (a \cup b) \right) \geq \sum_{i=1}^n \theta_{[0 \cdots i]} \cdot \theta_{[1 \cdots n]} \cdot p \cdot (p^\text{max} - p_{\alpha, \alpha}) \geq n \cdot \theta_{[0 \cdots n]} \cdot p \cdot \delta
\]
where we used \(\theta_{[0 \cdots n]} = \theta_{[0 \cdots i]} \cdot \theta_{[i \cdots n]}\) and that \(p^\text{max} - p_{\alpha, \alpha} \geq \delta\) (by definition of \(\delta\) and as \(\alpha \notin \text{Act}^\text{max}(t)\)).

Furthermore, the \(T\)-paths \(\zeta \in \text{pref}(\omega')\) that do not belong to \(\text{pref}(\omega)\) are extensions of \(\omega\) of the form \(\omega \alpha \zeta'\). Their cumulative \(\text{en}\)-value is:
\[
\sum_{\zeta \in \text{pref}(\omega')} \Pr_S^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta) = \Pr_S^\omega(\omega) \cdot p \cdot \text{en}_\alpha\omega,\alpha
\]
Using the fact that \(\Pr_S^\omega(\omega)\) equals \(\theta_{[0 \cdots n]}\) and that
\[
\sum_{\zeta \in \text{pref}(\omega')} \Pr_S^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta) \geq \sum_{\zeta \in \text{pref}(\omega)} \Pr_S^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta)
\]
we obtain:
\[
\text{en}_\alpha\omega,\alpha - \text{en}_\alpha\omega = \sum_{\zeta \in \text{pref}(\omega')} \Pr_S^\omega(\zeta) \cdot \Pr_T^\omega_{\text{last}}(\zeta)
\]
We have \(n \geq K\) (as \(\omega\) is a path of \(\Pi_s\) and by definition of \(\Pi_t\)). Moreover, \(K \cdot \delta \geq e\) (by the choice of \(K\)). We conclude:
\[
n \cdot \delta \geq e
\]
and therefore:
\[
\text{en}_\alpha\omega,\omega - \text{en}_\alpha\omega \geq n \cdot \theta_{[0 \cdots n]} \cdot p \cdot (e - \text{en}_\alpha\omega,\alpha)
\]
This completes the proof of (**).

With \(q = \theta_{[0 \cdots n]} \cdot p\), we obtain by (*) and (**):
\[
\text{en}_\alpha\omega,\omega \leq \text{en}_\omega,\omega + q \cdot (e - \text{en}_\alpha\omega,\alpha)
\]
\[
\text{en}_\alpha\omega,\omega \geq \text{en}_\alpha\omega,\omega + q \cdot (e - \text{en}_\alpha\omega,\omega)
\]
and therefore:
\[
\text{LP}_M^\omega (a \cup b) = \frac{\text{en}_\alpha\omega,\omega + q \cdot (e - \text{en}_\alpha\omega,\omega)}{\text{en}_\alpha\omega,\omega + q \cdot (e - \text{en}_\alpha\omega,\omega)}
\]
We now use (\(\dagger\)) and (\(\ddagger\)) to show that \(\text{LP}_M^\omega (a \cup b) \geq \text{LP}_M^\omega (a \cup b)\).

Obviously, the expected number of steps is an upper bound for the expected accumulated probability for \(a \cup b\). In particular:
\[
eap_{t_{\alpha,s}} \leq \text{en}_t_{\alpha,s}
\]
Hence, if \(e = \text{en}_t_{\alpha,s}\) then \(e - \text{en}_t_{\alpha,s} \geq 0\) and therefore:
\[
\text{LP}_M^\omega (a \cup b) \geq \frac{\text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s}} = \text{LP}_M^\omega (a \cup b)
\]
Suppose now that \(e \neq \text{en}_t_{\alpha,s}\).
If \(e > \text{en}_t_{\alpha,s}\) then
\[
\frac{e - \text{en}_t_{\alpha,s}}{e - \text{en}_t_{\alpha,s}} \geq 1 \geq \frac{\text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s}}
\]
We now use the fact that if \(x, y, z, w\) are non-negative real numbers with \(w, y > 0\) then:
\[
\frac{x + y}{y + w} \geq \frac{x}{y} \iff \frac{x}{y} \geq \frac{x}{y}
\]
This yields:
\[
\text{LP}_M^\omega (a \cup b) \geq \frac{\text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s}} = \text{LP}_M^\omega (a \cup b)
\]
It remains to consider the case \(e < \text{en}_t_{\alpha,s}\). Here, we use the fact that for all non-negative rational numbers \(x, y, z, w\) with \(y > w > 0\) then:
\[
\frac{x - z}{y - w} \geq \frac{x}{y} \iff \frac{x}{y} \leq \frac{x}{y}
\]
In particular, if \(x \leq y\) and \(0 < z < y\) then \((x - z)/(y - w) \leq x/y\). Hence:
\[
\frac{e - \text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s} - e} \leq \frac{\text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s}}
\]
which again yields:
\[
\text{LP}_M^\omega (a \cup b) \geq \frac{\text{en}_t_{\alpha,s}}{\text{en}_t_{\alpha,s} - e} = \text{LP}_M^\omega (a \cup b)
\]
For the general case where \(\Pi_t\) is a (countable and prefix-free) set of paths \(\omega\), for which \(\omega\) schedules several actions not in \(\text{Act}^\text{max}\), the argument is fairly the same. The essential difference is that we have to consider all state-action pairs \((t, \alpha)\) in \(B^\omega\) such that state \(t = (t, y)\) is reachable from \(s\) via a path in \(\Pi_s\) and \(\alpha \in \text{Act}(t) \setminus \text{Act}^\text{max}(t)\) such that \(\omega\) schedules action \(\alpha\) for \(t\) with positive probability. So, the lower bound for \(\text{LP}_M^\omega (a \cup b)\) in (\(\ddagger\)) then has the form:
\[
\text{LP}_M^\omega (a \cup b) \geq \frac{\text{en}_t_{\alpha,s} + \sum_{t, \alpha} q_{t, \alpha} \cdot (e - \text{en}_t_{\alpha,s})}{\text{en}_t_{\alpha,s} + \sum_{t, \alpha} q_{t, \alpha} \cdot (e - \text{en}_t_{\alpha,s})}
\]
where \(q_{t, \alpha}\) is the product of the probability in \(B^\omega\) of reaching \(t\) from \(s\) along a path in \(\Pi_s\) and the probability for taking action \(\alpha\) in state \(t\) of \(B^\omega\).
**Max-lrp via max-MP**

We now turn to the soundness of the constructed MDP $\mathcal{K}$ that encodes a counter for the number of consecutive $A$-states that have been visited in a suffix of the current path.

Let us observe that $\mathcal{K}$ might be not strongly connected, even if $\mathcal{M}$ is strongly connected. The reason is that not all states in $\mathcal{K}$ might be reachable from the states $s_{\mathcal{K}}$. If, however, $\mathcal{M}$ is strongly connected and we restrict $\mathcal{K}$ to the states reachable from some states $s_{\mathcal{K}}$ then $\mathcal{K}$ is strongly connected. This justifies to drop the starting state when talking maximal long-run probabilities or the maximal expected mean payoff in $\mathcal{K}$.

We start with the observation that the switch from $\mathcal{M}$ to $\mathcal{K}$ preserves the maximal long-run probabilities for $\alpha U b$.

**Lemma A.14.** $\text{LPP}_{\mathcal{M}}(\alpha U b) = \text{LPP}_{\mathcal{K}}(\alpha U b)$

**Proof.** obvious from Lemma IV.7. \qed

The remaining task is to show that $\text{LPP}_{\mathcal{M}}(\alpha U b)$ is the maximal expected mean payoff in $\mathcal{K}$. For this, we switch from $\mathcal{K}$ to the following MDP $N$ which uses state- rather than transition-based weights. The definition of the weighted MDP $N$ is as follows. The state space of $N$ is:

$$S_N = C \times \{0\} \cup (A \cup B) \times \{1, \ldots, K, T\}$$

The new MDP $N$ has the same action set as $M$ and the following transition probabilities:

- If $s \in C$ then $P_N((s,0), \alpha, (s',k)) = P(s, \alpha, s')$ if either $s' \in C \land k = 0$ or $s' \in A \cup B \land k = 1$.
- If $(s,i) \in A \times \{1, \ldots, K-1\}$ then:
  - $P_N((s,k), \alpha, (s',0)) = P(s, \alpha, s')$ if $s' \in C$
  - $P_N((s,k), \alpha, (s',1+i)) = P(s, \alpha, s')$ if $s' \in A \cup B$
- If $s \in A$, $k \in \{K, T\}$ and $\alpha \in \text{Act}_{\mathcal{K}}(s)$ then:
  - $P_N((s,k), \alpha, (s',0)) = P(s, \alpha, s')$ if $s' \in C$
  - $P_N((s,k), \alpha, (s',1)) = P(s, \alpha, s')$ if $s' \in A \cup B$
- If $s \in B$ and $k \in \{1, \ldots, K, T\}$ then:
  - $P_N((s,k), \alpha, (s',0)) = P(s, \alpha, s')$ if $s' \in C$
  - $P_N((s,k), \alpha, (s',1)) = P(s, \alpha, s')$ if $s' \in A \cup B$

And $P_N(\cdot) = 0$ in all remaining cases. The weight function of $N$ assigns rational values to states:

- $\text{wgt}_N(s,k) = k$ if $s \in B$ and $k \in \{1, \ldots, K\}$
- $\text{wgt}_N(s,k) = p_s^{\text{max}}. K$ if $s \in A$
- $\text{wgt}_N(s,T) = p_s^{\text{max}}$ if $s \in A \cup B$

Note that for $s \in B$ we have $\text{wgt}_N(s,T) = 1$.

As for $\mathcal{K}$, with the above definition $N$ might be not strongly connected. If, however, we restrict $N$ to the fragment that is reachable from the state $s_N$ then $N$ is strongly connected.

**Lemma A.15.** $\text{LPP}_{\mathcal{N}}^{\text{max}}(\alpha U b) = \text{LPP}_{\mathcal{N}}^{\text{max}}(\alpha U b)$ and $\text{E}_{\mathcal{N}}^{\text{max}}(\text{MP}) = \text{E}_{\mathcal{N}}^{\text{max}}(\text{MP})$

**Proof.** obvious as $N$ can be viewed as a variant of $K$ with state-based (rather than transition-based) weights. \qed

Hence, the remaining is to prove that $\text{LPP}_{\mathcal{N}}^{\text{max}}(\alpha U b)$ coincides with $\text{E}_{\mathcal{N}}^{\text{max}}(\text{MP})$.

**Lemma A.16.** For each FM-scheduler $\mathcal{G}$ for $N$:

$$\text{E}_{N,s_N}^{\mathcal{G}}(\text{MP}) \geq \text{LPP}_{N,s_N}^{\mathcal{G}}(\alpha U b)$$

Furthermore, if $\mathcal{G}$ leaves $A \times \{T\}$ with probability 1 whenever it reaches this set of states, i.e., if the Markov chain induced by $\mathcal{G}$ has no BSCC consisting of states in $A \times \{T\}$, then equality holds.

**Proof.** Let $\mathcal{G}$ be an FM-scheduler for $N$ and let $\mathcal{C} = e^{\mathcal{G}}$ be the induced finite Markov chain with $S_C = S_N \times X$ where $X$ denotes the set of modes of $\mathcal{G}$. Thus, the states of $C$ are triples $(s,k,x)$ with $s \in S$, $k \in \{0,1,\ldots,K,T\}$ and $x \in X$ where $k = 0$ if $s \in C$ and $k \in \{1,\ldots,K,T\}$ if $s \in A \cup B$.

It suffices to prove that the expected mean payoff of each BSCC $B$ of $C$ is greater or equal than the long-run probability for $\alpha U b$ in $B$. In what follows, we fix a BSCC $B$ of $C$ and use the following notations:

- Symbol $s$ is used to denote states in $B$.
- $\theta_s$ denotes the long-run frequency (steady-state probability) of state $s$ in $B$.
- $p_s$ denotes the probability for $\alpha U b$ in $C$ (resp. $B$) from state $s$.
- $w_s$ denotes the weight of state $s$ in $C$, i.e., if $s = (s,k,x)$ then $w_s = \text{wgt}_N(s,k)$.

For sets of $A$- and $B$-states inside the given BSCC $B$ we shall use the following notations:

$$\mathcal{A} = B \cup (A \times \{1,\ldots,K\} \times X)$$
$$\mathcal{B} = B \cup (B \times \{1,\ldots,K\} \times X)$$

We write $\mathcal{A}_T$ for $\mathcal{A}$ extended by the set of states $(s,T,x) \in B$ with $s \in A$. Similarly, $\mathcal{B}_T$ is $\mathcal{B}$ extended by the states $(s,T,x) \in B$ with $s \in B$.

Treating $\theta = (\theta_s)_{s \in S_C}$ as a row vector and $p = (p_s)_{s \in S_C}$ and $w = (w_s)_{s \in S_C}$ as column vectors:

$$\text{E}_{N,s_N}^{\mathcal{G}}(\text{MP}) - \text{LPP}_{N,s_N}^{\mathcal{G}}(\alpha U b) = \langle \theta, w - p \rangle$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product. Using the fact that $w_s = p_s = 1$ for each state $s = (s,k,x)$ in
Let \( s \in B \) and \( k = \top \) and \( w_s = p_s = 0 \) for each state \( s = (s, 1, x) \) in \( C \) with \( s \in C \) we obtain:

\[
\langle \theta, w - p \rangle = \sum_{s \in C} \theta_s \cdot (w_s - p_s) = \sum_{s \in A \cup B \cup C} \theta_s \cdot (w_s - p_s) \sum_{s \in \Theta \cup \emptyset} \theta_s \cdot (w_s - p_s)
\]

If \( \Theta \) leaves \( A \times \{\top\} \) with probability 1 after entering, then equality holds, as then \( w_s = p_s = p_s^{\max} \) for each state \( s = (s, k, x) \) in \( B \) with \( s \in A \) and \( k = \top \).

We define the matrix

\[
\mathbf{P} \in [0, 1]^{A \cup B}
\]

as the transition probability matrix of \( C \) restricted to the states in \( A \cup B \) and with all outgoing transitions from states in \( B \) and \( A \times \{k\} \) removed, i.e., the rows with index \((s, k, x)\) where \( s \in B \) or \( k = K \) are set to 0. Let \( \theta' \), \( p' \) and \( w' \) be the vectors \( \theta \), \( p \) and \( w \) projected to \( A \cup B \).

Furthermore, let \( \theta'[i] \) be the vector obtained from \( \theta' \) by setting all entries not indexed by \((s, i, x), s \in S, x \in X, i \in 0 \) and define \( p'[i] \) analogously. Let now \( i, j \) be integers with \( K \geq j \geq i \geq 1 \). Then:

\[
\theta'[j] = \theta'[i] \cdot \mathbf{P}^{j-i} \quad \text{and} \quad \theta' = \sum_{i=0}^{K-1} \theta'[i] \cdot \mathbf{P}^{i}.
\]

Intuitively, the left equation states that we can compute the steady state probabilities for states with index \( j \) using only the steady state probabilities for states with index \( i \) and the \( j-i \) steps transition probability matrix \( \mathbf{P}^{j-i} \). This follows from the fact that states indexed by \( j \) can only be reached from states indexed by \( i \) in \( j-i \) steps.

We now regard the vector \( d \in [0, 1]^{A \cup B} \) with \( d(s, k, x) = 1 \) if \( s \in B \), \( d(s, K, x) = p_s^{\max} \), and \( d(s, k, x) = 0 \) in all other cases. Then:

\[
p'[i] \leq \sum_{j=0}^{K-i} \mathbf{P}^{j} \cdot d
\]

where the inequality is understood componentwise. Again, equality holds if \( \Theta \) always leaves \( A \times \{\top\} \) almost surely.

So, we get:

\[
\sum_{s \in A \cup B} \theta_s \cdot p_s = \sum_{k=1}^{K} (\theta'[k], p'[k]) \leq \sum_{k=1}^{K} (\theta'[k], \mathbf{P}^{k-1}) \cdot d
\]

But now, we see that the weight structure in \( N \) was just defined such that the last line sums up to \( \langle \theta', w' \rangle \).

Thus:

\[
\sum_{s \in A \cup B} \theta_s \cdot w_s = \sum_{s \in A \cup B} \theta_s \cdot w_s
\]

We conclude that \( \langle \theta, w - p \rangle \geq 0 \) and \( \langle \theta, w - p \rangle = 0 \) if \( \Theta \) leaves \( A \times \{\top\} \) almost surely.

This yields \( E_B^\Theta(MP) \geq E_B^\Theta(aUb) \) in the general case and \( E_B^\Theta(MP) = E_B^\Theta(aUb) \) if \( \Theta \) leaves \( A \times \{\top\} \) almost surely.

**Lemma A.17.** \( E_N^{\max}(MP) = E_N^{\max}(aUb) \)

**Proof.** Thanks to Lemma A.16, it suffices to construct a scheduler \( \Sigma \) for \( N \) such that \( L_\Sigma^T_N(s_N, aUb) = E_N^{\max}(MP) \) for all states \( s_N \).

Let \( \Theta \) be an MD-scheduler for \( N \) that maximizes the expected mean payoff from every state in \( N \). We may assume w.l.o.g. that the induced Markov chain (restricted to the states that are reachable from the states \( s_N \)) has a single BSCC, say \( B \).

Case 1: \( B \) is not contained in \( A \times \{\top\} \). Then \( \Theta \) always leaves \( A \times \{\top\} \) with probability 1. The second part of Lemma A.16 yields:

\[
E_N^{\max}(MP) = E_N^{\Theta}(MP) = L_\Sigma^T_N(s_N, aUb)
\]

Hence, we can deal with \( \Theta = \Sigma \).

Case 2: \( B \) is contained in \( A \times \{\top\} \), In this case, we can construct an infinite-memory scheduler \( \Sigma \) with \( L_\Sigma^T_N(s_N, aUb) = E_N^{\Theta}(MP) \). The idea is that scheduler \( \Sigma \) stays inside \( B \) for larger and larger number of steps and in between it leaves \( B \) to maximize the probability for \( aUb \). In this way, the weight of states in \( B \) is equal to the probability of \( aUb \) under \( \Sigma \) and the long-run probability then equals this probability. \( \square \)

**Corollary A.18 (See Lemma IV.8).** \( L_\Sigma^{\max}(aUb) = E_N^{\max}(MP) \).
Proof. The claim follows by combining the results that have been established so far:

\[
\begin{align*}
\mathbb{LP}^\text{max}_M(a \cup b) &= \mathbb{LP}^\text{max}_X(a \cup b) \quad \text{(Lemma A.14)} \\
&= \mathbb{LP}^\text{max}_N(a \cup b) \quad \text{(Lemma A.15)} \\
&= \mathbb{EP}^\text{max}_N(\text{MP}) \quad \text{(Lemma A.17)} \\
&= \mathbb{EP}^\text{max}_X(\text{MP}) \quad \text{(Lemma A.15)} \\
\end{align*}
\]

\(\square\)

**NP-hardness**

We now turn to the threshold problem for long-run probabilities and until-properties that takes as input a strongly connected \(M\), atomic propositions \(a, b\) and a rational value \(\delta\) and asks whether \(\mathbb{LP}^\text{max}_M(a \cup b) \geq \delta\). By the results that have been established so far, this problem belongs to EXPTIME. We now provide an NP lower bound.

**Theorem A.19.** The threshold problem “is \(\mathbb{LP}^\text{max}_M(a \cup b) \geq \delta\)” is NP-hard.

**Proof.** We prove the statement by a polynomial reduction from the intersection problem for unary DFA, i.e., DFA over a one-letter alphabet. This problem is known to be NP-complete [5].

So, we are given a finite number of unary DFA, say \(D_1, \ldots, D_k\) over the alphabet \(\Sigma = \{0\}\), where \(D_i = (Q_i, \Sigma, \delta_i, q_{0i}, F_i)\). We simply write \(\delta_i(q)\) rather than \(\delta_i(q, 0)\). We may suppose the transition functions \(\delta_i\) are total and that \(Q_i \cap Q_j = \emptyset\) if \(i \neq j\). W.l.o.g. we further assume that \(|Q_i| \geq 2\) for all \(i \leq k\).

We are going to construct an MDP \(M\) over \(AP = \{a, b\}\) and a rational value \(\delta\) such that \(\mathbb{LP}^\text{max}_M(a \cup b) \geq \delta\) if and only if \(\mathcal{L}(D_1) \cap \cdots \cap \mathcal{L}(D_k)\) is nonempty. Obviously, the latter is equivalent to the statement that there exists some \(n \in \mathbb{N}\) with \(n < \ell\) and \(0^n \in \mathcal{L}(D_1) \cap \cdots \cap \mathcal{L}(D_k)\) where \(\ell = |Q_1| \cdots |Q_k|\).

Let \(A = (Q, \Sigma, \delta, Q_0, F)\) denote the NFA resulting from the union of \(D_1, \ldots, D_k\). That is, \(Q = Q_1 \cup \ldots \cup Q_k\), \(Q_0 = \{q_{01}, \ldots, q_{0k}\}\), \(F = F_1 \cup \ldots \cup F_k\) and \(\delta(q) = \delta_i(q)\) if \(q \in Q_i\). That is, besides the nondeterministic choice of the initial state, \(A\) behaves deterministically. The idea is now to treat \(A\) as a substructure of an MDP \(M\) where all states of \(M\) are labeled by \(a\). The substructure \(A\) can be entered in \(M\) via an action that assigns probabilities \(1/k\) to each of the \(k\) initial states, possibly using a “pumping option” to increase the number of \(a\)-states before simulating \(M\). In each state \(q \in Q\) an action \(\alpha\) is enabled that mimics \(A\)’s transition from \(q\) with probability \((r-1)/r\) and moves back to the state where \(A\) can be entered (in which neither \(a\) nor \(b\) holds) with probability \(1/r\). For the final states \(q \in F\) there is also an \(\beta\) that leads with probability \(1\) to a \(b\)-state form which \(M\) can return to the state where \(M\) can be entered. Other states and transitions of \(M\) are needed for technical purposes.

The idea of this construction is as follows. To maximize the long-run probability for \(a \cup b\) in \(M\), a scheduler guesses a natural number \(n\) with \(n \leq \ell\) and \(0^n \in \mathcal{L}(D_1) \cap \cdots \cap \mathcal{L}(D_k)\). It then uses the pumping option to enter \(M\) after having generated a sequence of \(\ell - n\) \(a\)-states. It then attempts to make \(n\) steps inside \(A\) using action \(\alpha\) and leaves \(A\) using the \(\beta\) transition.

Formally, the state space of \(M\) is

\[
S = Q \cup \{a, b, c, init\}
\]

where the states in \(Q \cup \{a\}\) are labeled by \(a\) and \(b\) is labeled by \(b\). The action set is \(Act = \{\alpha, \beta, enter, pump, \tau\}\). The transition probabilities are as follows (where \(r\) and \(r'\) are rational numbers \(> 1\) defined later):

- In \(s \in \{init, a\}\), actions \(enter\) and \(pump\) are enabled with the transition probabilities:
  
  \[
  P(s, enter, q_{0i}) = \frac{r-1}{k \times r}, \quad i = 1, \ldots, k,
  \]
  \[
  P(s, enter, init) = \frac{1}{r}.
  \]
  
- \(P(s, pump, a) = \frac{r-1}{r}, \quad P(s, pump, init) = \frac{1}{r}\).

- In each state \(q \in Q\), action \(\alpha\) is enabled with:
  
  \[
  P(q, \alpha, \delta(q)) = \frac{r-1}{r}, \quad P(q, \alpha, init) = \frac{1}{r}.
  \]

For the final states \(q \in F\), additionally action \(\beta\) is enabled with \(P(q, \beta, b) = 1\).

- In \(s' \in \{b, c\}\), action \(\tau\) is enabled with:
  
  \[
  P(s', \tau, c) = \frac{r'-1}{r'}, \quad P(s', \tau, init) = \frac{1}{r'}.
  \]

**Scheduler \(S_n\) and its long-run probability.** Before defining the values \(r\) and \(r'\), let us suppose \(n\) is an integer with \(n < \ell\) and \(0^n \in \mathcal{L}(D_1) \cap \cdots \cap \mathcal{L}(D_k)\). Regard the FMD-scheduler \(S_n\) which behaves as follows. Let \(m = \ell - n - 1\). From state \(init\), \(S_n\) attempts to reach \(a\) and stay there via the pumping option \(pump\) until the generated path ends in \(m\) consecutive visits to \(a\). Afterwards the scheduler \(S_n\) enters \(A\) via \(enter\). Let \(q_{0i}\) be the state reached after that sequence. \(S_n\) then attempts to follow the unique accepting run for \(0^n\) in \(D_i\) (via action \(\alpha\)). If successful it reaches a final state \(q \in F_i\). It then takes action \(\beta\) and returns to \(init\) via the \(\tau\)-transitions in states \(b\) and \(c\). Of course, each of the attempts might fail, in which case \(S_n\) returns to \(init\) and behaves in the same way.

The long-run probability \(\mathbb{LP}^\text{max}_N(a \cup b)\) under \(S_n\) can be computed as the quotient of expected accumulated probability and expected number of steps between two visits to \(init\): Let \(\rho = \frac{r-1}{r}\). For \(1 \leq k \leq \ell\), the scheduler succeeds in generating a path starting with \(k\) consecutive \(a\)-states with probability \(\rho^k\). The probability to satisfy \(aU b\) afterwards is \(\rho^{\ell-k}\). In addition \(b\) is
reached with probability \( \rho^\ell \) and here the probability of \( aU b \) is 1. So, the expected accumulated probability is
\[
\left( \sum_{k=1}^{\ell} \rho^k \cdot \rho^{\ell-k} \right) + \rho^\ell = \rho^\ell (\ell + 1).
\]
For the expected number of steps, we get the following: The probability to return to \( init \) in exactly \( 1 \leq k \leq \ell \) steps is given by \( (1 - \rho)\rho^{k-1} \). With probability \( \rho^\ell \), the state \( b \) is reached in which case the expected return time is \( \ell + 1 + r' \). All in all, the expected number of steps is
\[
\sum_{k=1}^{\ell} k(1 - \rho)\rho^{k-1} + (\ell + 1 + r')\rho^\ell = \frac{1 - \rho^\ell}{1 - \rho} - \ell \cdot \rho^\ell + (\ell + 1 + r')\rho^\ell = \frac{r + (r' + 1 - r)\rho^\ell}{1 - \rho}
\]
as \( \frac{1}{1 - \rho} = r \). We let \( r' := r - 1 \) and obtain
\[
\mathbb{E}[\tau_{\text{max}}(aUb)] = \frac{\rho^\ell (\ell + 1)}{r} = (1 - \rho)\rho^\ell (\ell + 1).
\]

**Definition of \( \tau \) (and \( r := (r-1)/r \)):** For \( j \in \mathbb{N} \), define \( \rho_j := (1 - \rho)\rho^j (j + 1) \). The goal is to choose \( r \) in such a way that \( \varphi_j > \varphi_i \) for all \( j \in \mathbb{N} \setminus \{ \ell \} \). In particular, this means that we have to choose \( r \) such that \( \rho^j (j + 1) \) reaches a maximum for \( j = \ell \). This is achieved by \( \tau = \ell + 1.5 \) as can be seen as follows. Let \( j \in \mathbb{N} \) and \( \vartheta \in \{ <, =, > \} \). Then:
\[
\rho^j (j + 1) \vartheta \rho^{j+1} (j + 2) \iff \frac{j + 1}{j + 2} \vartheta \frac{1}{\rho}
\]
In particular, if \( \tau = \ell + 1.5 \) then \( \ell + 2 > \tau \) and therefore:
\[
\varphi_j > \varphi_{j+1} > \varphi_{j+2} > \ldots
\]
On the other hand, \( (\ell - 1) + 2 = \ell + 1 < \tau \) and therefore:
\[
\varphi_j > \varphi_{j-1} > \varphi_{j-2} > \ldots
\]

**Definition of the threshold value \( \vartheta \):** The idea is to use the observation that the maximal long-run probability that can be achieved when the intersection language is empty is bounded by (maybe, needs to be checked):
\[
\vartheta := \frac{\varphi_j - \mu}{k}
\]
where \( \mu \) is the minimal difference between the value \( \varphi_j \) and one the values \( \varphi_i \) for \( j \neq \ell \).

Note that when entering \( A \) after having visited state \( a \) \( m \)-times, then at least one of the DFA \( D_1, \ldots, D_k \) does not accept the word \( 0^{\ell-m} \). And taking \( \alpha \) resp. \( pump \) fewer or more times than \( \ell \) (which means firing the \( \beta \)-transition to state \( b \) after having visited \( j \) \( a \)-states for some \( j \) different from \( \ell \)) yields a value \( \varphi_j < \varphi_{\ell} \). Same for combinations of such options depending on which of the states \( q_{0, \ell} \) is entered.

As the values \( \varphi_j \) are strictly increasing for \( j < \ell \) and strictly decreasing for \( j > \ell \) we have:
\[
\mu = \min \{ \varphi_{\ell} - \varphi_{\ell-1}, \varphi_{\ell} - \varphi_{\ell+1} \}
\]
Using the fact that
\[
\rho^\ell = (1 - \frac{1}{\ell})^\ell \geq 1 - \frac{\ell}{7}
\]
and using \( \tau = \ell + 1.5 \) and \( (1 - \rho) = 1/\rho \) we obtain:
\[
\varphi_{\ell} - \varphi_{\ell-1} = (1 - \rho)(\rho^\ell (\ell + 1) - \rho^{\ell-1} \ell) \geq \frac{1}{7}(1 - \frac{\ell}{7})(1 + \ell - \ell/\rho) = \frac{3}{4\ell^2} \cdot (1 - \frac{\ell}{\tau}) = \frac{3}{4\ell^2} \geq \frac{3}{4\ell^2}
\]
and
\[
\varphi_{\ell} - \varphi_{\ell+1} = (1 - \rho)(\rho^\ell (\ell + 1) - \rho^{\ell+1} (\ell + 2)) \geq \frac{1}{7}(1 - \frac{\ell}{7})(1 + \ell + 1 - \rho - 2\rho) = \frac{3}{4\ell^2} \cdot \frac{1}{7} = \frac{3}{4\ell^2}.
\]
This yields \( \mu \geq \frac{1}{4\ell^2} \). In addition, \( k \leq \log(\ell) \) as we assume that all \( D_i \) have at least two states. To get a safe approximation, we observe that for \( \ell \geq 1000 \), we have \( \frac{\ell}{\ell^2} < \frac{\ell}{\ell^2} \).

The crux is now to find a rational threshold \( \vartheta \) with \( \vartheta < \varphi \leq \vartheta \) that is computable in polynomial time. We cannot use \( \varphi_{\ell} \) as its logarithmic length is exponential in the input size (the sum of the sizes of \( D_1, \ldots, D_k \)). Instead, we will compute an approximation \( \varphi' \) of \( \varphi_{\ell} \cdot \ell^k \) up to an absolute error \( < 5 \) in polynomial time. Then for \( \ell \geq 1000 \), we can choose the threshold \( \vartheta \) to be \( \varphi'_{\ell} / 5 \) as
\[
\varphi_{\ell} > \varphi'_{\ell} - 50 \ell^4 \geq \varphi_{\ell} - 100 \ell^4 > \varphi_{\ell} - \mu/k = \vartheta_{\ell}.
\]

**Approximation of \( \varphi_{\ell} \cdot \ell^k \):** We define the real function
\[
f(z) := \frac{(1/z + 1/2)^{1/z}}{(1/z + 3/2)^{1/z}} (1/z + 1)
\]
for \( z \in [-1/2, 1/2] \setminus [0] \) and \( f(0) := \frac{1}{e} \) where \( e \) is Euler's number. The idea is that \( f(1/\ell) = \varphi_{\ell} \) and that we can provide good approximations of \( f(z) \) for \( z \) close to 0 using Taylor's theorem. By standard methods from calculus, we see that the function \( f \) is 5 times continuously differentiable on \([-1/2, 1/2]\). Calculating the derivatives, we obtain the following approximation:
\[
f(z) = \frac{1}{e} + \frac{z}{2e} - \frac{z^2}{3e} + \frac{z^3}{4e} - \frac{313z^4}{1440e} + O(z^5)
\]
for \( z \to 0 \). So there are reals \( c_0, z_0 > 0 \) such that
\[
|f(z) - \left( \frac{1}{e} + \frac{z}{2e} - \frac{z^2}{3e} + \frac{z^3}{4e} - \frac{313z^4}{1440e} \right)| \leq c_0 e z^5
\]
for \( |z| < z_0 \). Using \( \varphi_{\ell} = f(1/\ell) \) we obtain
\[
\varphi_{\ell} \ell^4 - \frac{1}{e} \left( \ell^4 + \frac{1}{2} \ell^3 - \frac{1}{3} \ell^2 + \frac{1}{4} \ell - \frac{313}{1440} \right) \leq c_0 \ell^{-1}
\]
for $\ell > 1/z_0$.

Let $L = \max(1/z_0, c_0, 1000)$. To obtain the desired approximation of $g_{\ell} \ell^4$ for $\ell > L$, it is enough to approximate

$$\frac{1}{e} \left( \ell^4 + \frac{1}{2} \ell^3 - \frac{1}{3} \ell^2 + \frac{1}{4} \ell - \frac{313}{1440} \right)$$

up to an absolute error of 49. Hence, approximating $\frac{1}{e}$ up to an absolute error of $\frac{1}{\ell^4}$ is sufficient. This can be done in polynomial time.

So, we can compute a threshold value for $\ell > L$ which completes the reduction of the intersection problem for unary DFA to the threshold problem for maximal long-run probabilities in MDPs. As $L$ is defined in terms of the function $f$, i.e. independent of all variables, and as there are only finitely many instances with $\ell \leq L$, this finishes the proof. \qed