# Long-run Satisfaction of Path Properties

Christel Baier[1], Nathalie Bertrand[2], Jakob Piribauer[1], Ocan Sankur[2]
[1]Technische Universität Dresden, Germany
[2]Univ 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 $a \mathrm{U}_q b$ of the until operator relaxes the standard meaning of $a \mathrm{U} b$ 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.

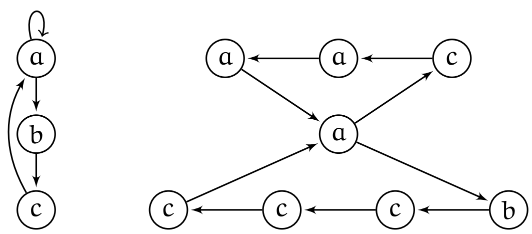


Fig. 1. Kripke structures requiring memory to maximize the long-run frequency of $a \mathrm{U} b$.

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 $a \mathrm{U} b$ 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 $abca^2bca^4bca^8bc\ldots$ 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 $a \mathrm{U} b$ is 1. The KS on the right illustrates, that, even when infinite-memory is not needed, memoryless is not enough: for $a \mathrm{U} b$, the maximal long-run frequency is achieved by alternating between the two simple cycles and amounts to $\frac{4}{9}$, which is indeed more than $\frac{2}{5}$ the

The authors are partly supported by the DFG through the Collaborative Research Centers CRC 912 (HAEC) and TRR 248 (see `https://perspicuous-computing.science`, project ID 389792660), the Cluster of Excellence EXC 2050/1 (CeTI, project ID 390696704, as part of Germany's Excellence Strategy), the Research Training Groups QuantLA (GRK 1763) and RoSI (GRK 1907), the DFG-project BA-1679/11-1, and the DFG-project BA-1679/12-1. The collaboration is supported by the Inria associated team QuantProb.

978-1-7281-3608-0/19/$31.00 

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 $\mathcal{N}_k$ shown in Fig. 2, the only non-determinism is between actions $\alpha$ and $\beta$, and $\alpha$ yields a uniform distribution over the three successors.

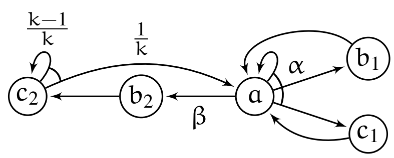


Fig. 2. MDP with labels indicated by the state names requiring counting to maximize the long-run probability of $a \, \mathrm{U} \, b$.

Under the memoryless scheduler $\mathfrak{S}_\alpha$ that always picks action $\alpha$, the probability of $a \, \mathrm{U} \, b$ in the $a$-state is $\frac{1}{2}$, and its frequency is $\frac{3}{5}$. The state $b_1$ has frequency $\frac{1}{5}$ and from there the probability of $a \, \mathrm{U} \, b$ is 1. We thus compute the long-run probability under $\mathfrak{S}_\alpha$ to be $\frac{1}{2}$. Similarly, the steady-state probability of the states $a$ and $b_2$ under the memoryless scheduler $\mathfrak{S}_\beta$ are $\frac{1}{k+2}$, and the probability that $a \, \mathrm{U} \, b$ holds from there is 1. The long-run probability of $a \, \mathrm{U} \, b$ under $\mathfrak{S}_\beta$ equals $\frac{2}{k+2}$. Observe that the satisfaction probability at the $a$-state depends on the scheduler (it is $\frac{1}{2}$ for $\mathfrak{S}_\alpha$ and 1 for $\mathfrak{S}_\beta$). 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 $a \, \mathrm{U} \, b$) 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 $\Box_q$ [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 $\Box_q$-modality, while long-run probabilities are a quantitative measure across behaviors. For finite strongly connected Markov chains, the probabilities for $\Box_q$-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 $\mathcal{M}$, $\Box_q(a \, \mathrm{U} \, b)$ holds in $\mathcal{M}$ with probability 1 iff the long-run probability of $a \, \mathrm{U} \, b$ 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 $\Box_q$-formulas. On the

| | *non-probabilistic case* Kripke structures | *probabilistic case* Markov decision processes |
|---|---|---|
| reachability/invariant/ Rabin/Streett conditions | value computable in polynomial-time (special case of Theorem IV.3) | value computable in polynomial-time (Theorem IV.3) |
| constrained reachability $(a \, \mathrm{U} \, b)$ | value computable in polynomial-time (Corollary III.3) | qualitative decision problems in polynomial-time (Lemma IV.5)<br>value computable in pseudo-polynomial time (Theorem IV.10)<br>NP-hard threshold problem (Theorem IV.11) |
| regular co-safety (NFA) | value computable in exponential time (Corollary III.6)<br>PSPACE-complete threshold problem (Theorem III.7) | computability of the value: open<br>PSPACE-hard threshold problem (consequence of Theorem III.7) |

TABLE I
SUMMARY OF THE MAIN RESULTS.

other hand, they cannot deal with formulas of the type $\Box_q(a \mathrm{U} b)$ for $q < 1$. The results in [23] only apply to $q = 1$. The fragment in [24] can deal with $\Box_q$-modalities for arbitrary q, but imposes the constraint that no until operator occurs in the scope of the $\Box_q$-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].

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

*Kripke structures (KSs):* A KS is a tuple $\mathcal{T} = (S, \Delta, \mathsf{AP}, \mathsf{L})$ where $(S, \Delta)$ is a finite directed graph, $\mathsf{AP}$ a finite set of atomic propositions and $\mathsf{L} : S \to 2^{\mathsf{AP}}$ a labeling function. The *trace* of a path $\pi = s_0\, s_1, s_2 \ldots$ is the word $\mathsf{L}(\pi) = \mathsf{L}(s_0)\,\mathsf{L}(s_1)\,\mathsf{L}(s_2)\ldots$ over $2^{\mathsf{AP}}$ 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 $\pi_{[i \ldots j]}$ for the path fragment $s_i\, s_{i+1} \ldots s_j$. Likewise, $\pi_{[0 \ldots i]}$ and $\pi_{[i \ldots]}$ stand for the prefix ending in state $s_i$ resp. the suffix starting from $s_i$. 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", $\Box T$ for the event "always T" and $\Box\Diamond T$ 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.*, $s \models \exists \Diamond T$ indicates that a T-state is reachable from state s.

*Weighted structures:* A *weighted KS* extends a plain KS $\mathcal{T}$ as above by a weight function $wgt : \Delta \to \mathbb{Q}$ 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 $mp(\pi)$ is defined by:

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

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

*Markov decision processes (MDPs):* An MDP is a tuple $\mathcal{M} = (S, Act, P)$ where S is a finite state space, *Act* a finite nonempty set of actions and $P : S \times Act \times S \to$

$[0,1]\cap\mathbb{Q}$ the transition probability function satisfying $\sum_{t\in S} P(s,\alpha,t) \in \{0,1\}$ for all $(s,\alpha) \in S\times Act$. The triples $(s,\alpha,t)$ with $P(s,\alpha,t)>0$ are called transitions of $\mathcal{M}$. The actions enabled in s form $Act(s)=\{\alpha\in Act: \sum_{t\in S} P(s,\alpha,t)=1\}$. We often refer to the pairs $(s,\alpha)$ with $\alpha\in Act(s)$ as the state-action pairs of $\mathcal{M}$. 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 (numerator and denominator of) all weights.

Intuitively, when $\mathcal{M}$ 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 $P(s,\alpha,\cdot)$. 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 $P(s_i,\alpha_i,s_{i+1}) > 0$ for all $i \geqslant 0$. The notations $first(\pi)$, $\pi_{[i]}$, $\pi_{[i\ldots j]}$ 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* for $\mathcal{M}$ is a function $\mathfrak{S}$ that maps a finite path $\varpi$ to a probability distribution over $Act(last(\varpi))$ (where $last(\varpi)$ is the last state of $\varpi$). Given a finite path $\varpi$, then $\mathfrak{S}\uparrow\varpi$ denotes the *residual scheduler* defined by $(\mathfrak{S}\uparrow\varpi)(\varpi')=\mathfrak{S}(\varpi;\varpi')$ for each finite path $\varpi'$ starting in $last(\varpi)$. $\mathfrak{S}$ is *deterministic* if $\mathfrak{S}(\varpi)$ is a Dirac distribution for each finite path, in which case $\mathfrak{S}$ 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.

$\Pr^{\mathfrak{S}}_{\mathcal{M},s}$ denotes the probability measure induced by $\mathfrak{S}$, 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 $\Pr^{\max}_{\mathcal{M},s}(\varphi)$ and $\Pr^{\min}_{\mathcal{M},s}(\varphi)$ for $\omega$-regular properties. We consider the maximal (resp. minimal) expected mean payoff, denoted $\mathbb{E}^{\max}_{\mathcal{M},s}(\mathbb{MP})$ and $\mathbb{E}^{\min}_{\mathcal{M},s}(\mathbb{MP})$, 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 $\Pr^{\max}_{\mathcal{E}}(\Diamond b)$ or $\mathbb{E}^{\max}_{\mathcal{E}}(\mathbb{MP})$ for ECs $\mathcal{E}$. As shown in [19], for each scheduler $\mathfrak{S}$ 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 $\mathcal{T}=(S,\Delta,\mathsf{AP},\mathsf{L})$ 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:

$$lrf_{\varphi}(\pi) \quad = \quad \liminf_{n\to\infty} \frac{1}{n+1}\cdot\sum_{i=0}^{n} \mathbb{1}_{\pi_{[i\ldots]}\models\varphi}$$

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

$$\mathbb{LF}^{\max}_{\mathcal{T},s}(\varphi) \;=\; \sup_{\pi} lrf_{\varphi}(\pi)$$

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

Obviously, the value $lrf_{\varphi}(\pi)$ 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 $\mathbb{LF}^{\max}_{\mathcal{T}}(\varphi)$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(\varphi)$.

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 prefix-independent and where a classification of cycles into good and bad ones is not sufficient. The example from Fig. 1 left, illustrates this phenomenon as already for apparently simple properties such as constrained reachability, maximizing the long-run frequency may require infinite memory. As a stepping stone in this direction, we consider until properties, and regular co-safety properties where satisfaction is witnessed by "good" prefixes. We first give a polynomial time algorithm for computing optimal long-run frequencies for until properties. The main contribution for long-run frequencies in KSs, is then the PSPACE-completeness for co-safety properties specified by NFA.

### 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 $\varphi = a \,\mathrm{U}\, b$ with $a, b \in \mathsf{AP}$. Let

$$\begin{aligned} A &= \{ s \in S \mid s \models \exists(a \,\mathrm{U}\, b), s \not\models \forall(a \,\mathrm{U}\, b) \}, \\ B &= \{ s \in S \mid s \models \forall(a \,\mathrm{U}\, b) \}, \quad C = S \setminus (A \cup B) \ . \end{aligned}$$

If $B = \varnothing$, $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) = \mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b) = 0$, thus we assume $B \neq \varnothing$. The until properties $a \,\mathrm{U}\, b$ and $A \,\mathrm{U}\, B$ are equivalent: for each infinite path $\pi$ of $\mathcal{T}$, $\pi \models a \,\mathrm{U}\, b$ iff $\pi \models A \,\mathrm{U}\, B$, and therefore $\mathit{lrf}_{a \mathrm{U} b}(\pi) = \mathit{lrf}_{A \mathrm{U} B}(\pi)$.

Using these sets, we first characterize easy cases:

**Lemma III.1.** $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) = 1$ *iff* $\mathcal{T}$ *has a* $A \cup B$*-cycle.* $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b) = 0$ *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 $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b)$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b)$ are respectively 1 and 0. This observation permits the additional assumption that $\mathcal{T}$ has no $A$-cycles.

The computation of $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b)$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b)$ 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 $A \,\mathrm{U}\, B$ 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,* $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\max}_{\mathcal{T}'}$ *and* $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\min}_{\mathcal{T}'}$.

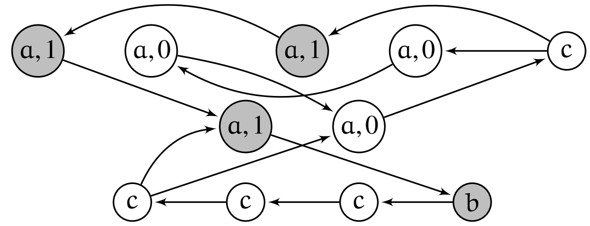


Fig. 3. Weighted Kripke structures obtained for the KS right of Fig. 1: gray states have weight 1, while others have weight 0.

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* $\mathbb{LF}^{\max}_{\mathcal{T},s}(a \,\mathrm{U}\, b)$ *and* $\mathbb{LF}^{\min}_{\mathcal{T},s}(a \,\mathrm{U}\, b)$ *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 $A \,\mathrm{U}\, B$. (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 $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ be an NFA over the alphabet $\Sigma = 2^{\mathsf{AP}}$ representing $\varphi$, *i.e.*, an infinite path of $\mathcal{T}$ satisfies $\varphi$ iff it has a prefix accepted by $\mathcal{A}$. Hence, $\mathit{lrf}_{\varphi}(\pi)$, also denoted $\mathit{lrf}_{\mathcal{A}}(\pi)$, is the long-run average of positions in $\pi$ where a word in $\mathcal{L}(\mathcal{A})$ starts, and we write $\mathbb{LF}^{*}_{\mathcal{T}}(\mathcal{A})$ rather than $\mathbb{LF}^{*}_{\mathcal{T}}(\varphi)$. We show that the computation of $\mathbb{LF}^{\max}_{\mathcal{T}}(\mathcal{A})$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(\mathcal{A})$ reduces to determine the maximal and minimal mean-payoff in a weighted KS $\mathcal{G}$ 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 $\mathcal{A}$. We also assume that $q_0 \notin F$ (otherwise $\mathcal{A}$ 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 $\mathcal{G}$ as follows. Let $\ell = |Q|$ denote the number of states in the NFA $\mathcal{A}$. Then, the state space $S_{\mathcal{G}}$ of $\mathcal{G}$ is equal to

$$S \times \big\{ \{\emptyset\} \cup \{\mathsf{merged}\} \times \{0, \ldots, \ell\} \cup Q \times \{\mathsf{true}, \mathsf{false}\} \big\}^{2\ell+1}.$$

Each state of $\mathcal{G}$ tracks a state of $\mathcal{T}$ and in addition a vector of $2\ell+1$ elements, which we explain now. For a state $(s, f) \in S_{\mathcal{G}}$, $f(i)$ 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; $(q, \mathsf{true})$ means that the track is at state $q$ of $\mathcal{A}$ and that the final state of $\mathcal{A}$ will be eventually reached (confirming the guess that $\varphi$ holds), while $(q, \mathsf{false})$ means that no final state of $\mathcal{A}$ should be reached at this track (expressing the guess that $\varphi$ does not hold). The symbol $\mathsf{merged}$ is used to merge tracks that arrive at the same states of $\mathcal{A}$ with the same guess. The formal details follow.

The set $I_{\mathcal{G}}$ of initial states in $\mathcal{G}$ is given by:

$$I_{\mathcal{G}} \;=\; \Big\{\; (s_0, f) \;\mid\; \begin{array}{l} f(0) \in \{(q_0, \mathsf{true}), (q_0, \mathsf{false})\} \\ \text{and } f(i) = \emptyset \text{ for } 1 \leqslant i \leqslant 2\ell \end{array} \;\Big\}$$

That is, a single track is created at initial state $q_0$ of $\mathcal{A}$ 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 $\mathcal{A}$. Note that for each track of the form $(q, \mathsf{true})$, the construction will guess *some* successor $q'$ of $q$ in $\mathcal{A}$ in an attempt to prove the guess correct, while for each track of the form $(q, \mathsf{false})$, it will check *all* successors of $q$ in order to prove that the word that is being read is not accepted by $\mathcal{A}$.

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 $\mathcal{G}$ and each state $s'$ of $\mathcal{T}$, we define relation $\rightsquigarrow_1$ as follows: we have $(s, f) \rightsquigarrow_1 (s', f', H)$ where $H = \bigcup_{q:\exists i.f(i)=(q,\mathsf{false})} \delta(q, L(s))$, if, and only if the following holds for all $0 \leqslant i \leqslant 2\ell$,

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

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

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

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

Assume $(s', f', H)$ is consistent. Let us write $H' = \{q \in H \mid \forall i. f'(i) \neq (q, \mathsf{false})\} = \{q'_1, \ldots, q'_k\}$. We define the relation $\rightsquigarrow_2$ by $(s', f', H) \rightsquigarrow_2 (s', f'')$ if, and only if, the following conditions hold for all $0 \leqslant i \leqslant 2\ell$:

Case $f'(i) = \emptyset$: if $i$ is the first such position then $f''(i) = (q_0, b)$ for some $b \in \{\mathsf{true}, \mathsf{false}\}$, if it is the $l+1$-th such position, we let $f''(i) = q'_l$, and otherwise $f''(i) = \emptyset$.

Case $f'(i) = (q, b)$ for some $(q, b) \in Q \times \{\mathsf{true}, \mathsf{false}\}$: if for all $0 \leqslant j < i$, $f'(j) \neq (q, b)$, then $f''(i) = (q, b)$; and otherwise, $f''(i) = (\mathsf{merged}, j)$ 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 $(\mathsf{merged}, j)$ 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 $(q, \mathsf{false})$, and a new track is started from $q_0$.

The transitions of $\mathcal{G}$ are defined as follows. We have $((s, f), (s', f'')) \in \Delta_{\mathcal{G}}$ if and only if $(s, s') \in \Delta$ and there exists a consistent state $(s', f', H)$ such that $(s, f) \rightsquigarrow_1 (s', f', H) \rightsquigarrow_2 (s'', f'')$.

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

For each $0 \leqslant i \leqslant 2\ell$, let us define the following labels. Let $F_i$, $\mathsf{false}_i$, $\mathsf{merged}_i$, and $\emptyset_i$ denote the states $(s, f)$ satisfying, respectively, $f(i) = (q, \mathsf{true})$ for some $q \in F$, $f(i) = (q, \mathsf{false})$ for some $q \in Q$, $f(i) = (\mathsf{merged}, j)$ for some $0 \leqslant j < i$, 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:

$$\Phi \quad = \quad \bigwedge_{i=0}^{2\ell} \Box\Diamond\big(\mathsf{false}_i \vee \mathsf{merged}_i \vee F_i \vee \emptyset_i\big).$$

Let $\mathbb{MP}^{\max}_{\mathcal{G}}(\Phi)$ denote the maximal mean payoff in $\mathcal{G}$ starting in one of the two initial states in $I_{\mathcal{G}}$ while satisfying $\Phi$. Similarly, $\mathbb{MP}^{\min}_{\mathcal{G}}(\Phi)$ 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, $\mathbb{LF}^{\max}_{\mathcal{T}}(\mathcal{A}) = \mathbb{MP}^{\max}_{\mathcal{G}}(\Phi)$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(\mathcal{A}) = \mathbb{MP}^{\min}_{\mathcal{G}}(\Phi)$.*

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 $(q, \mathsf{true})$ must eventually either end in an accepting state $(F_i)$ 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 $(\cdot, \mathsf{false})$ or not be used $(\emptyset_i)$. 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.** *$\mathbb{LF}^{\max}_{\mathcal{T},s}(\varphi)$ and $\mathbb{LF}^{\min}_{\mathcal{T},s}(\varphi)$ are computable in time exponential in the size of $\mathcal{A}$ and polynomial in the size of $\mathcal{T}$.*

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 $\mathcal{T}$, an NFA $\mathcal{A}$ and a rational threshold $\vartheta$, the threshold problem asks whether $\mathcal{T}$ admits an infinite path $\pi$ with $\mathit{lrf}_{\mathcal{A}}(\pi) \geqslant \vartheta$.

**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 $s_{\mathcal{G}}$ in $\mathcal{G}$ and two cycles $\xi_{\Phi}$ and $\xi_{\mathrm{MP}}$ containing $s_{\mathcal{G}}$ and satisfying certain length restrictions and $\xi_{\Phi}^{\omega} \models \Phi$ and $\mathit{mp}(\xi_{\mathrm{MP}}^{\omega}) \geqslant \vartheta$. In fact, one can construct the desired path by alternating between $\xi_{\Phi}$ and $\xi_{\mathrm{MP}}$ 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. $\square$

**Proposition III.8 (Qualitative thresholds).**
*Deciding the existence of an infinite path $\pi$ in $\mathcal{T}$ with*

- $\mathit{lrf}_{\mathcal{A}}(\pi) > 0$ *is in PTIME;*
- $\mathit{lrf}_{\mathcal{A}}(\pi) = 0$ *(resp. $= 1$) is NP-hard;*
- $\mathit{lrf}_{\mathcal{A}}(\pi) < 1$ *is PSPACE-hard.*

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

To prove the two NP-hardness results, we provide a polynomial reduction from 3SAT. The construction of the KS $\mathcal{T}$ 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 $\mathit{lrf}_{\mathcal{A}}(\pi) = 1$, and use the fact that PSPACE is closed under complementation. The reduction is easy: the automaton is the one from the universality instance, and the Kripke structure generates all possible sequences of letters. See Lemma A.8 for the full proofs. $\square$

## 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 $a \,\mathrm{U}\, b$). 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 $\mathcal{M} = (S, \mathit{Act}, \mathsf{P}, \mathsf{AP}, \mathsf{L})$ and a path property $\varphi$. The *long-run probability* for $\varphi$ of an infinite path $\pi$ under a scheduler $\mathfrak{S}$ for $\mathcal{M}$ is defined as as the long-run average of the probabilities for $\varphi$ in all positions of $\pi$ with respect to the residual schedulers $\mathfrak{S}\uparrow\pi_{[0\ldots i]}$ (see Section II for the definition of these):

$$\mathit{lrp}_{\varphi}^{\mathfrak{S}}(\pi) \quad = \quad \liminf_{n\to\infty} \frac{1}{n+1}\cdot\sum_{i=0}^{n} \Pr_{\mathcal{M},\pi_{[i]}}^{\mathfrak{S}\uparrow\pi_{[0\ldots i]}}(\varphi) \ .$$

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

$$\begin{aligned}\mathbb{LP}_{\mathcal{M},s}^{\max}(\varphi) &= \sup_{\mathfrak{S}} \ \mathbb{LP}_{\mathcal{M},s}^{\mathfrak{S}}(\varphi)\\ \mathbb{LP}_{\mathcal{M},s}^{\min}(\varphi) &= \inf_{\mathfrak{S}} \ \mathbb{LP}_{\mathcal{M},s}^{\mathfrak{S}}(\varphi)\end{aligned}$$

where $\mathfrak{S}$ ranges over all schedulers for $\mathcal{M}$, 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 $\mathit{lrp}_{\varphi}^{\mathfrak{S}}$, depends on the scheduler $\mathfrak{S}$.

**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 $\mathfrak{S}$-path then $\Pr_{\mathcal{M},\pi_{[i]}}^{\mathfrak{S}\uparrow\pi_{[0\ldots i]}}(\varphi) = \mathbb{1}_{\pi_{[i\ldots]}\models\varphi}$ . Hence, $\mathit{lrf}_{\varphi}(\pi) = \mathit{lrp}_{\varphi}^{\mathfrak{S}}(\pi)$, and therefore $\mathbb{LF}_{\mathcal{M},s}^{\max}(\varphi) = \mathbb{LP}_{\mathcal{M},s}^{\max}(\varphi)$ and $\mathbb{LF}_{\mathcal{M},s}^{\min}(\varphi) = \mathbb{LP}_{\mathcal{M},s}^{\min}(\varphi)$.

Analogously to extremal long-run frequencies in KSs, we can assume MDPs to be strongly connected. Indeed for a general MDP $\mathcal{M}$ without traps, one can compute the MECs $\mathcal{E}_1,\ldots,\mathcal{E}_k$ in polynomial-time [13], [19]. Given a path formula $\varphi$, we define a weighted MDP $\mathcal{N}$, obtained from $\mathcal{M}$ by collapsing each MEC $\mathcal{E}_i$ 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 $\mathbb{LP}_{\mathcal{E}_i}^{\max}(\varphi)$ to state $s_i'$ and 0 to all other states. With this construction, $\mathbb{LP}_{\mathcal{M},s}^{\max}(\varphi)$ equals the maximal expected accumulated weight in $\mathcal{N}$ 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 $\mathcal{M}$ is strongly connected, the optimal long-run probabilities do not depend on the starting state and we simply write $\mathbb{LP}_{\mathcal{M}}^{\max}(\varphi)$ and $\mathbb{LP}_{\mathcal{M}}^{\min}(\varphi)$.

### 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 $\mathit{lrp}_{\varphi}(\pi)$ for each infinite path $\pi$. If $\varphi$ is an $\omega$-regular property, for each bottom strongly connected component (BSCC) $\mathcal{B}$ of the Markov chain $\mathcal{M}$, the long-run probability for all states in $\mathcal{B}$ is the same:

$$\mathbb{LP}_{\mathcal{B}}(\varphi) \quad = \quad \sum_{t\in\mathcal{B}} \mathit{lrf}_{\mathcal{B}}(t)\cdot\Pr_{\mathcal{M},t}(\varphi)$$

where $\mathit{lrf}_{\mathcal{B}}(t)$ denotes the steady-state probability (defined as the long-run frequency) of state $t$ in $\mathcal{B}$. Thus, $\mathbb{LP}_{\mathcal{B}}(\varphi)$ equals the probability for $\varphi$ in $\mathcal{B}$ viewed as a Markov chain where the initial distribution is given by the long-run frequencies in $\mathcal{B}$, which again coincides with the expected mean payoff in $\mathcal{B}$ when $\Pr_{\mathcal{M},t}(\varphi)$ is viewed as weight for state $t$. The long-run frequencies inside the BSCC are computable in polynomial-time using a linear equation system. The values $\Pr_{\mathcal{M},t}(\varphi)$ 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, $\mathbb{LP}_{\mathcal{M},s}(\varphi)$ is computable in polynomial time for those properties $\varphi$ where $\Pr_{\mathcal{M},t}(\varphi)$ 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 $\mathit{wgt}(s) = \Pr_{\mathcal{B},s}(\varphi)$ to each state $s$ can also be written as a quotient of expectations as follows. Let $s$ be an arbitrary state in $\mathcal{B}$, called *reference state*. Then, the long-run probability for $\varphi$ in $\mathcal{B}$ 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,

$$\mathbb{LP}_{\mathcal{B}}(\varphi) \ = \ \frac{\mathbb{E}_{\mathcal{B},s}(\text{“weight until } s\text{”})}{\mathbb{E}_{\mathcal{B},s}(\text{“steps until } s\text{”})} \qquad (\star)$$

Finally, if $\mathfrak{B}$ denotes the set of all BSCCs of $\mathcal{M}$ then for each state $s$ in $\mathcal{M}$:

$$\mathbb{LP}_{\mathcal{M},s}(\varphi) \quad = \quad \sum_{\mathcal{B}\in\mathfrak{B}} \Pr_{\mathcal{M},s}(\Diamond\mathcal{B})\cdot\mathbb{LP}_{\mathcal{B}}(\varphi) \ .$$

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 $\mathfrak{S}$ exist that maximize the probabilities for $\varphi$ from every visited state in the sense that

$$\Pr^{\mathfrak{S}\uparrow\pi_{[0...i]}}_{\mathcal{M},\pi_{[i]}}(\varphi) \;=\; \Pr^{\max}_{\mathcal{M},\pi_{[i]}}(\varphi)$$

for each infinite $\mathfrak{S}$-path $\pi$ and each position $i \in \mathbb{N}$. Obviously, such a scheduler $\mathfrak{S}$ maximizes the long-run probability for $\varphi$. (See Theorem A.9 and its proof.)

**Theorem IV.3 (Efficiently solvable lrp-instances).** *The values $\mathbb{LP}^{\max}_{\mathcal{M},s}(\varphi)$ and $\mathbb{LP}^{\min}_{\mathcal{M},s}(\varphi)$ are computable in polynomial-time if $\varphi$ has the form $\Diamond b$ (reachability), $\Box b$ (invariant), $\bigwedge_{i=1}^{n}\bigvee_{j=1}^{\ell_i}(\Box\Diamond b_{i,j} \wedge \Diamond\Box a_{i,j})$ (generalized Rabin condition), or $\bigwedge_{i=1}^{n}(\Box\Diamond a_i \to \Box\Diamond b_i)$ (Streett condition). Moreover, FMD-scheduler 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 $a \,\mathrm{U}\, b$. While long-run frequencies for $a \,\mathrm{U}\, b$ 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 $\mathcal{M}$ (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 $\mathfrak{S}_\alpha$ and $\mathfrak{S}_\beta$ achieve long-run probabilities of $1/2$ and $\frac{2}{k+2}$ for the until property $a \,\mathrm{U}\, b$. Thus, as soon as $k \geqslant 3$, $\mathfrak{S}_\alpha$ achieves the higher long-run probability, whereas $\mathfrak{S}_\beta$ maximizes the probability of $a \,\mathrm{U}\, b$ from the $a$-state. However, none of them is optimal. To see this, consider the FMD-schedulers $\mathfrak{T}_n$ for $n \geqslant 1$, 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, $\mathfrak{T}_n$ schedules action $\alpha$ if the counter value is at most $n$ and $\beta$ otherwise.

We compute $\mathbb{LP}^{\mathfrak{T}_n}_{\mathcal{N}_k}(a \,\mathrm{U}\, b)$ via the quotient representation shown in ($\star$) in Remark IV.2 using that the Markov chain $\mathcal{C}_n$ induced by $\mathfrak{T}_n$ 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 $(a,i)$ means state $a$ with counter value $i$. 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 $\mathit{ef}(s)$ of the states $s$ in $\mathcal{C}_n$ along the return paths from and to $(a,1)$. These values are: $\mathit{ef}(a,i) = \frac{1}{3^{i-1}}$ for $i = 1, \ldots, n{+}1$, $\mathit{ef}(b_1) = \mathit{ef}(c_1) = \left(1 - \frac{1}{3^n}\right) \cdot \frac{1}{2}$, $\mathit{ef}(b_2) = \frac{1}{3^n}$, and $\mathit{ef}(c_2) = \frac{k}{3^n}$.

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:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}_n,(a,1)}(\text{“steps until } (a,1)\text{”}) \\
&= \sum_{i=1}^{n+1} \tfrac{1}{3^{i-1}} \;+\; \left(1 - \tfrac{1}{3^n}\right)\cdot\tfrac{1}{2}\cdot 2 \;+\; \tfrac{1}{3^n} \;+\; \tfrac{k}{3^n} \\
&= \tfrac{1}{4}\cdot\left(10 + (4k-2)\cdot\tfrac{1}{3^n}\right)
\end{aligned}$$

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

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}_n,(a,1)}(\text{“weight until } (a,1)\text{”}) \\
&= \sum_{s} \mathit{ef}(s) \cdot \Pr_{\mathcal{C}_n,s}(a \,\mathrm{U}\, b)
\end{aligned}$$

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

$$\begin{aligned}
&\mathbb{E}_{\mathcal{C}_n,(a,1)}(\text{“weight until } (a,1)\text{”}) \\
&= \sum_{i=1}^{n+1} \tfrac{1}{3^{i-1}}\cdot\tfrac{1}{2}\cdot\left(1 + \tfrac{1}{3^{n-i+1}}\right) \;+\; \left(1 - \tfrac{1}{3^n}\right)\cdot\tfrac{1}{2}\cdot 1 \\
&= \tfrac{1}{4}\cdot\left(5 + (2n+3)\cdot\tfrac{1}{3^n}\right), \text{ and}
\end{aligned}$$

$$\mathbb{LP}^{\mathfrak{T}_n}_{\mathcal{N}_k}(a \,\mathrm{U}\, b) = \frac{5 + (2n+3)\cdot\frac{1}{3^n}}{10 + (4k-2)\cdot\frac{1}{3^n}}.$$

Thus, $\mathfrak{T}_n$ achieves a higher long-run probability than $\mathfrak{S}_\alpha$ if $n > k{-}2$. To determine which scheduler is optimal among the schedulers $\mathfrak{T}_n$ with $n \in \mathbb{N}$, we determine the least natural number $n$ such that $\mathbb{LP}^{\mathfrak{T}_n}_{\mathcal{N}_k}(a \,\mathrm{U}\, b) > \mathbb{LP}^{\mathfrak{T}_{n+1}}_{\mathcal{N}_k}(a \,\mathrm{U}\, b)$. Treating the computed expression for $\mathbb{LP}^{\mathfrak{T}_n}_{\mathcal{N}_k}(a \,\mathrm{U}\, b)$ 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 $k \geqslant 2$, we obtain $n = k-1$.

We will see later that the maximal long-run probability of $\mathcal{N}_k$ is indeed achieved by $\mathfrak{T}_n$ for this $n$ (see Remark IV.12 below). Note that $\mathcal{N}_k$ has 5 states and its size is in $\mathcal{O}(\log k)$. 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

$$\mathbb{LP}^{\mathfrak{T}_{k-1}}_{\mathcal{N}_k}(a \,\mathrm{U}\, b) = \frac{5 \cdot 3^{k-1} + 2k + 1}{10 \cdot 3^{k-1} + 4k - 2}$$

the greatest common divisor of enumerator and denominator is at most 4 (note that $2(5 \cdot 3^{k-1} + (2k+1)) - (10 \cdot 3^{k-1} + (4k-2)) = 4$). 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 $a \,\mathrm{U}\, b$. We first fix useful notations. Given a state $s$ of $\mathcal{M}$, let $p_s^{\max} = \mathrm{Pr}^{\max}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$, $p_s^{\min} = \mathrm{Pr}^{\min}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$ and:

$$\begin{aligned} A &= \{ s \in S \mid p_s^{\max} > 0 \text{ and } p_s^{\min} < 1 \}, \\ B &= \{ s \in S \mid p_s^{\min} = 1 \}, \quad C = S \setminus (A \cup B). \end{aligned}$$

Then, $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) = \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, B)$ for every $s$ and $\mathfrak{S}$. 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 $\mathfrak{S}$ such that $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$ 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): $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) > 0$ for some scheduler $\mathfrak{S}$ iff $B \neq \varnothing$. For the value 1 problem, we can rely on: $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 1$ iff $\exists \mathfrak{S}. \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) = 1$ iff $\mathcal{M}$ has a $(A \cup B)$-EC $\mathcal{E}$ with $\mathcal{E} \cap B \neq \varnothing$ or $\mathcal{M}$ has an $A$-EC $\mathcal{E}$ with $\mathrm{Pr}^{\max}_{\mathcal{E}}(a \,\mathrm{U}\, b) = 1$. 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):

$$\mathbb{LP}^{\min}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 1 - \mathbb{LP}^{\max}_{\mathcal{M}}(A \,\mathrm{U}\, C)$$

if $\mathcal{M}$ has no $A$-EC. Otherwise $\mathbb{LP}^{\min}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 0$.

For the rest of this section, we suppose that $\mathcal{M}$ is strongly connected and does not have $(A \cup B)$-EC $\mathcal{E}$ with $\mathcal{E} \cap B \neq \varnothing$, as otherwise $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ is 1.

The first important result is that $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ can be approximated by finite-memory schedulers. This is proved using Fatou's lemma (see Lemma A.12).

**Lemma IV.6.** $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) = \sup_{\mathfrak{S} \in \mathrm{FM}} \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$.

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 $\mathfrak{S}$ can be "improved" into another FM-scheduler if it does not maximize the probability for $a \,\mathrm{U}\, b$ (which does not require memory) after having generated a sequence of $K$ consecutive $a$-states. Here, improving means increasing the long-run probability of $a \,\mathrm{U}\, b$. We will show that such a saturation point $K$ is computable in time polynomial in the size of $\mathcal{M}$.

For $\alpha \in \mathit{Act}(s)$, let $p_{s,\alpha} = \sum_{t \in S} P(s,\alpha,t) \cdot p_t^{\max}$ and we write $\mathit{Act}^{\max}(s)$ for the set of *maximizing actions*, *i.e.* actions $\alpha \in \mathit{Act}(s)$ with $p_{s,\alpha} = p_s^{\max}$.

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

1) $\mathrm{Pr}^{\mathfrak{S}\uparrow\varpi}_{\mathcal{M},\mathit{last}(\varpi)}(a \,\mathrm{U}\, b) = p^{\max}_{\mathit{last}(\varpi)}$ for each finite $\mathfrak{S}$-path $\varpi$ that has a suffix consisting of $K$ or more consecutive $A$-states, and
2) the Markov chain induced by $\mathfrak{S}$ has a single BSCC.

By definition, any $\mathfrak{S} \in \mathrm{FM}(K)$ only schedules maximizing actions (in $\mathit{Act}^{\max}(\cdot)$) 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 $\mathfrak{S}$, written $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M}}(a \,\mathrm{U}\, b)$, (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:*

$$\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) = \sup_{\mathfrak{S} \in \mathrm{FM}(K)} \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M}}(a \,\mathrm{U}\, b) \ .$$

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

$$K \;=\; \max\{ |A|, \lceil e/\delta \rceil \} \qquad (*)$$

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

$$\delta_s \ = \ \min\big\{\, p_s^{\max} - p_{s,\alpha} : \alpha \in Act(s) \setminus Act^{\max}(s) \,\big\}$$

with the convention that $\min \varnothing = \infty$. 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 $\mathcal{M}$. Intuitively, each $\delta_s$ is the minimum probability loss when the first action of an optimal strategy for $a\,\mathrm{U}\,b$ 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 $\mathrm{FM}(K)$. Consider an MDP which arises from $\mathcal{M}$ 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 $Act^{\max}(\cdot)$) are enabled. Formally, given a positive integer $n$, we define the MDP $\mathcal{M}_n = (S_n, Act, P_n)$ 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:
  $P_n(s, \alpha, (s', 1)) = P(s, \alpha, s')$ if $s' \in A$
  $P_n(s, \alpha, s') = P(s, \alpha, s')$ if $s' \in B \cup C$
- If $(s, i) \in A \times \{1, \ldots, n{-}1\}$ then:
  $P_n((s, i), \alpha, s') = P(s, \alpha, s')$ if $s' \in B \cup C$
  $P_n((s, i), \alpha, (s', i{+}1)) = P(s, \alpha, s')$ if $s' \in A$
- If $s \in A$, $k \in \{n, \top\}$ and $\alpha \in Act^{\max}(s)$ then:
  $P_n((s, k), \alpha, s') = P(s, \alpha, s')$ if $s' \in B \cup C$
  $P_n((s, k), \alpha, (s', \top)) = P(s, \alpha, s')$ if $s' \in A$
- $P_n(\cdot) = 0$ in all remaining cases.

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

$$e_{t,s} \ = \ \mathbb{E}^{\min}_{\mathcal{M}_N, t_N}(\text{“steps until } s_N\text{”})$$

as the minimal expected number of steps from state $t_N$ to $s_N$ in $\mathcal{M}_N$, and define $e = \max\limits_{s,t \in S} e_{t,s}$. The values $e_{t,s}$ and a corresponding MD-scheduler $\mathfrak{R}_{N,s}$ for $\mathcal{M}_N$ 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 $\mathcal{M}_N$ is at most quadratic in the size of $\mathcal{M}$.

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

We define the MDP $\mathcal{K} = \mathcal{M}_K$ 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 $K \cdot p_s^{\max}$,
- transitions from states $(s, \top)$ where $s \in A$ have weight $p_s^{\max}$, and

the weight of all other transitions is 0.

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

**Lemma IV.8.** $\mathbb{LP}^{\max}_{\mathcal{M}}(a\,\mathrm{U}\,b) \ = \ \mathbb{E}^{\max}_{\mathcal{K}}(\mathbb{MP})$.

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

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

**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 $\mathcal{M}_n$ as well as the other notations as above. First, we compute the saturation point $K$: $\delta = p_a^{\max} - p_{a,\alpha} = \frac{1}{3}$. For the computation of $e$, we consider the MDP $\mathcal{M}_1$. As we only consider the states reachable from $(a, 1)$ and only action $\beta$ is available in this state in $\mathcal{M}_1$, 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 $\mathcal{K} = \mathcal{M}_9$ extended with the weight structure described above.

For $0 \leqslant n \leqslant 8$, let $\mathfrak{T}_n$ be the scheduler which chooses $\alpha$ in state $(a, i)$ if $i \leqslant n$ 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

$$\begin{aligned} \mathbb{E}^{\mathfrak{T}_n}_{\mathcal{K}}(\mathbb{MP}) \ &= \ \frac{\sum_{\ell=1}^{n} \frac{1}{3^\ell}(\ell+1) + \frac{1}{3^n}(n+2)}{2\sum_{\ell=1}^{n} \frac{1}{3^\ell}(\ell+1) + \frac{1}{3^n}(n+3)} \\ &= \ \frac{5 + (2n+3)\frac{1}{3^n}}{10 + \frac{2}{3^n}}. \end{aligned}$$

The maximum is obtained for $n = 1$. As some MD-scheduler maximizes the expected mean payoff in $\mathcal{K}$, we conclude that $\mathbb{E}^{\max}_{\mathcal{K}}(\mathbb{MP}) = \mathbb{E}^{\mathfrak{T}_1}_{\mathcal{K}}(\mathbb{MP}) = \frac{5}{9}$. This value matches the value $\mathbb{LP}^{\mathfrak{T}_1}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ computed in Example IV.4. □

The length of the binary representation of $K$ is polynomial in the size of $\mathcal{M}$, so that the size of $\mathcal{K} = \mathcal{M}_K$ is pseudo-polynomial in the size of $\mathcal{M}$. 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 $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ and $\mathbb{LP}^{\min}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ 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 $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) \geqslant \vartheta$, 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 $\mathcal{D}_1, \ldots, \mathcal{D}_k$ be unary DFAs, and let $\ell$ be the product of their numbers of states. Then, $\mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$ is nonempty if and only if $\mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$ contains a word of length at most $\ell$.

We construct an MDP $\mathcal{M}$ with the disjoint union $\mathcal{A}$ of these DFAs as a substructure and a threshold $\vartheta$ 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 $\mathcal{M}$ are the $\mathcal{A}$-states and four additional states *init*, $a, b, c$. States in $\mathcal{A}$ 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 $\mathcal{A}$ are available. In each state in $\mathcal{A}$ an action $\alpha$ following the transition in the DFAs is available. All of these actions may fail with a small probability of $\frac{1}{r}$. Failure leads back to *init*. In the final states in $\mathcal{A}$, an action $\beta$ leading to $b$ is available. In $b$ and $c$ an action $\tau$ is available which leads to *init* with probability $\frac{1}{r'}$ and to $c$ otherwise.

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

**Remark IV.12 (Memory requirements).** If $\mathcal{M}$ does not have an $A$-EC then there are FMD-schedulers achieving the optimal long-run frequency for $a \,\mathrm{U}\, b$ 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 $\mathcal{K}$ 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 $\mathcal{M}$ containing some $B$-state also contains a $C$-state and there is an $A$-EC $\mathcal{E}$ where $\Pr^{\max}_{\mathcal{E}}(a \,\mathrm{U}\, b) > p^{\max}_s$ for all $A$-states $s$ that do not belong to some $A$-EC. □

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

- *if $\mathcal{M}$ has no $A$-cycle, or*
- *if for every scheduler $\mathfrak{S}$ and every $A$-state $s$, $\Pr^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) = p^{\max}_s$.*

*Proof.* In the first case indeed, we can let $K = |A|$ as saturation point, in which case the size of $\mathcal{K} = \mathcal{M}_K$ is only polynomial in the size of $\mathcal{M}$. In the second case, $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ is simply the maximal mean payoff in $\mathcal{M}$ 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


[1] Christel Baier, Nathalie Bertrand, Clemens Dubslaff, Daniel Gburek, and Ocan Sankur. Stochastic shortest paths and weight-bounded properties in Markov decision processes. In *33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 86–94. ACM, 2018.

[2] Christel Baier, Marcus Größer, and Frank Ciesinski. Quantitative analysis under fairness constraints. In *7th International Symposium on Automated Technology for Verification and Analysis (ATVA)*, volume 5799 of *Lecture Notes in Computer Science*, pages 135–150. Springer, 2009.

[3] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. MIT Press, 2008.

[4] Dimitri P. Bertsekas and John N. Tsitsiklis. An analysis of stochastic shortest path problems. *Mathematics of Operations Research*, 16(3):580–595, 1991.

[5] Michael Blondin, Andreas Krebs, and Pierre McKenzie. The complexity of intersecting finite automata having few final states. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:90, 2012.

[6] Udi Boker, Krishnendu Chatterjee, Thomas A. Henzinger, and Orna Kupferman. Temporal specifications with accumulative values. In *Proc. of the 26th Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 43–52. IEEE Computer Society, 2011.

[7] Benedikt Bollig, Normann Decker, and Martin Leucker. Frequency linear-time temporal logic. In *6th International Symposium on Theoretical Aspects of Software Engineering (TASE)*, pages 85–92. IEEE Computer Society, 2012.

[8] Patricia Bouyer, Nicolas Markey, and Raj Mohan Matteplackel. Averaging in LTL. In *25th International Conference on Concurrency Theory (CONCUR)*, volume 8704 of *Lecture Notes in Computer Science*, pages 266–280. Springer, 2014.

[9] Tomás Brázdil, Václav Brozek, Krishnendu Chatterjee, Vojtech Forejt, and Antonín Kucera. Two views on multiple mean-payoff objectives in markov decision processes. *Logical Methods in Computer Science*, 10(1), 2014.

[10] Pavol Cerný, Thomas A. Henzinger, and Arjun Radhakrishna. Simulation distances. *Theoretical Computer Science*, 413(1):21–35, 2012.

[11] Krishnendu Chatterjee and Laurent Doyen. Energy and mean-payoff parity Markov decision processes. In *Proc. of the 36th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 6907 of *Lecture Notes in Computer Science*, pages 206–218. Springer, 2011.

[12] Krishnendu Chatterjee, Andreas Gaiser, and Jan Kretínský. Automata with generalized Rabin pairs for probabilistic model checking and LTL synthesis. In *25th International Conference on Computer Aided Verification (CAV)*, volume 8044 of *Lecture Notes in Computer Science*, pages 559–575. Springer, 2013.

[13] Krishnendu Chatterjee and Monika Henzinger. Faster and dynamic algorithms for maximal end-component decomposition and related graph problems in probabilistic verification. In *22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1318–1336. SIAM, 2011.

[14] Krishnendu Chatterjee, Thomas A. Henzinger, and Marcin Jurdzinski. Mean-payoff parity games. In *20th Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 178–187, June 2005.

[15] Hana Chockler, Orna Kupferman, and Moshe Y. Vardi. Coverage metrics for temporal logic model checking*. *Formal Methods in System Design*, 28(3):189–212, 2006.

[16] Hana Chockler and Ofer Strichman. Before and after vacuity. *Formal Methods in System Design*, 34(1):37–58, 2009.

[17] Edmund Clarke, Orna Grumberg, and Doron Peled. *Model Checking*. MIT Press, 2000.

[18] Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

[19] Luca de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, Department of Computer Science, 1997.

[20] Luca de Alfaro. How to specify and verify the long-run average behavior of probabilistic systems. In *13th Annual IEEE Symposium on Logic in Computer Science (LICS)*, pages 454–465. IEEE Computer Society, 1998.

[21] Luca de Alfaro. Computing minimum and maximum reachability times in probabilistic systems. In *10th International Conference on Concurrency Theory (CONCUR)*, volume 1664 of *Lecture Notes in Computer Science*, pages 66–81, 1999.

[22] Andrzej Ehrenfeucht and Jan Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, Jun 1979.

[23] Vojtech Forejt and Jan Krcál. On frequency LTL in probabilistic systems. In Luca Aceto and David de Frutos-Escrig, editors, *26th International Conference on Concurrency Theory (CONCUR)*, volume 42 of *LIPIcs*, pages 184–197. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

[24] Vojtech Forejt, Jan Krcál, and Jan Kretínský. Controller synthesis for MDPs and frequency $\mathrm{LTL}_{\setminus\mathrm{GU}}$. In Martin Davis, Ansgar Fehnker, Annabelle McIver, and Andrei Voronkov, editors, *20th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR)*, volume 9450 of *Lecture Notes in Computer Science*, pages 162–177. Springer, 2015.

[25] Dror Fried, Solomon Eyal Shimony, Amit Benbassat, and Cenny Wenner. Complexity of Canadian traveler problem variants. *Theoretical Computer Science*, 487:1–16, 2013.

[26] Thomas A. Henzinger and Jan Otop. From model checking to model measuring. In *24th International Conference on Concurrency Theory (CONCUR)*, volume 8052 of *Lecture Notes in Computer Science*, pages 273–287. Springer, 2013.

[27] Lodewijk C.M. Kallenberg. Linear programming and finite Markovian control problems. *Mathematical Center Tracts*, 148, 1983.

[28] Richard M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23(3):309 – 311, 1978.

[29] Dexter Kozen. Lower bounds for natural proof systems. In *18th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 254–266. IEEE Computer Society, 1977.

[30] Orna Kupferman, Wenchao Li, and Sanjit A. Seshia. A theory of mutations with applications to vacuity, coverage, and fault tolerance. In *Formal Methods in Computer-Aided Design (FMCAD)*, pages 1–9. IEEE, 2008.

[31] Orna Kupferman and Moshe Y. Vardi. Robust satisfaction. In *10th International Conference on Concurrency Theory (CONCUR)*, volume 1664 of *Lecture Notes in Computer Science*, pages 383–398. Springer, 1999.

[32] Orna Kupferman and Moshe Y. Vardi. Vacuity detection in temporal model checking. *STTT*, 4(2):224–233, 2003.

[33] Albert R. Meyer and Larry J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *13th Annual Symposium on Switching and Automata Theory (SWAT/FOCS)*, pages 125–129. IEEE Computer Society, 1972.

[34] Christos H. Papadimitriou. Games against nature. *Journal of Computer and System Science*, 31(2):288–301, 1985.

[35] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, 1994.

[36] Paulo Tabuada and Daniel Neider. Robust linear temporal logic. In *25th EACSL Annual Conference on Computer Science Logic (CSL)*, volume 62 of *LIPIcs*, pages 10:1–10:21. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[37] Christian von Essen and Barbara Jobstmann. Synthesizing systems with optimal average-case behavior for ratio objectives. In *Proc. of the International Workshop on Interactions, Games and Protocols (iWIGP)*, volume 50 of *EPTCS*, pages 17–32, 2011.

## 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) $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) = 1$
 (2) $\mathcal{T}$ *has an infinite path* $\pi$ *with* $lrf_{a \mathrm{U} b}(\pi) = 1$.
 (3) $\mathcal{T}$ *has a* $A \cup B$*-cycle.*

(b) *The following three statements are equivalent:*
 (1) $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b) = 0$
 (2) $\mathcal{T}$ *has an infinite path* $\pi$ *with* $lrf_{a \mathrm{U} b}(\pi) = 0$.
 (3) $\mathcal{T}$ *has a* $A \cup C$*-cycle.*

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

"(1) $\Longrightarrow$ (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 of number of states then along every path latest after $n$ steps a state in $S \setminus (A \,\mathrm{U}\, B)$ will be reached. Hence, the long-run frequency of each infinite path is less or equal $(n-1)/n$. But then $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) \leqslant (n-1)/n < 1$

"(3) $\Longleftarrow$ (2)": suppose that $\mathcal{T}$ has a 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 contained in $A$. Let $\xi = s_0 s_1 \ldots s_n$ and let $\varpi = t_0 t_1 \ldots t_m t_{m+1} \ldots t_k$ be a path with $t_0 = s_0 = s_n = t_k$, $t_1, \ldots, t_{m-1} \in A$, $t_m \in B$. (Recall that by definition of $A$ we have $s \models \exists (A \,\mathrm{U}\, B)$ for all states $s \in A$.) Then, the long-run frequency of the infinite path $\xi; \varpi; \xi^2; \varpi; \xi^4; \varpi; \xi^8; \varpi; \ldots$ is 1.

"(2) $\Longrightarrow$ (3)": obvious.

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

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 $(s, s') \in \Delta$ is a transition in $\mathcal{T}$ where $s \in B \cup C$ then $\mathcal{T}'$ contains the following transitions:
  - $s \to (s', 0)$ and $s \to (s', 1)$ if $s' \in A$
  - $s \to s'$ if $s' \in B \cup C$
- If $(s, s') \in \Delta$ is a transition in $\mathcal{T}$ where $s \in A$ then $\mathcal{T}'$ has the following transitions:
  - $(s, 0) \to s'$ if $s' \in C$
  - $(s, 1) \to s'$ if $s' \in B$
  - $(s, 0) \to (s', 0)$ and $(s, 1) \to (s', 1)$ 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,* $\mathbb{LF}^{\max}_{\mathcal{T},s}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\max}_{\mathcal{T}', I(s)}$ *and* $\mathbb{LF}^{\min}_{\mathcal{T},s}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\min}_{\mathcal{T}', I(s)}$.

In particular, if $\mathcal{T}$ is strongly connected then $\mathbb{LF}^{\max}_{\mathcal{T}}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\max}_{\mathcal{T}'}$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(a \,\mathrm{U}\, b) = \mathbb{MP}^{\min}_{\mathcal{T}'}$ as stated in Lemma III.2.

*Proof.* We show that there is a bijection $\iota$ between the infinite paths in $\mathcal{T}$ and in $\mathcal{T}'$ such that for each infinite path $\pi$ in $\mathcal{T}$: $first(\iota(\pi)) \in I(first(\pi))$ and $lrf_{a \mathrm{U} b}(\pi) = mp(\iota(\pi))$.

Given an infinite path $\pi = s_0, s_1 s_2 \ldots$ in $\mathcal{T}$, we define $\iota(\pi) = s'_0 s'_1 s'_2 \ldots$ as follows. If $s_i \in B \cup C$ then $s'_i = s_i$. If $i = 0 \vee s_{i-1} \in B \cup C$ and $s_i s_{i+1} \ldots s_k \in A^* B$ then $s'_i s'_{i+1} \ldots s'_{k-1} = (s_i, 1)(s_{i+1}, 1) \ldots (s'_{k-1}, 1)$. Likewise, if $i = 0 \vee s_{i-1} \in B \cup C$ and $s_i s_{i+1} \ldots s_k \in A^* C$ then $s'_i s'_{i+1} \ldots s'_{k-1} = (s_i, 0)(s_{i+1}, 0) \ldots (s'_{k-1}, 0)$. It is easy to see that $\iota(\pi)$ is indeed a path in $\mathcal{T}'$ satisfying $first(\iota(\pi)) \in I(first(\pi))$ and $lrf_{a \mathrm{U} b}(\pi) = mp(\iota(\pi))$. Moreover, $\iota$ is a bijection as $\mathcal{T}$ has no $A$-cycles (and hence cannot stay forever in the sub-structures consisting of states in $A \times \{0,1\}$) and as $A \times \{0\}$ can only be left vis a transition to a $C$-state and $A \times \{1\}$ can only be left vis a transition to a $B$-state. $\square$

### 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 enough empty tracks to insert tracks at states $H'$ and one at $q_0$ at the second step.

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

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

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

Consider any path $(s_k, f_k)_{k \geqslant 0}$. The properties are true in the initial state $(s_0, f_0)$ by definition. Assume this holds at $(s_k, f_k)$. Consider $(s_{k+1}, f'_k, H)$ such that

$$(s_k, f_k) \rightsquigarrow_1 (s_{k+1}, f'_k, H) \rightsquigarrow_2 (s_{k+1}, f_{k+1}),$$

so that $(s_{k+1}, f'_k, H)$ 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 $Q \times \{\mathsf{true}, \mathsf{false}\}$ in $f_k$ are set to $\emptyset$; while each track of the form $(q, b) \in Q \times \{\mathsf{true}, \mathsf{false}\}$ is replaced by $(q', b)$ for some successor $q'$ of $q$ in $\mathcal{A}$. Thus, $|\{0 \leqslant i \leqslant 2\ell \mid f'_k(i) \neq \emptyset\}| \leqslant \ell$. All properties then follow from this observation. In fact, in $f_{k+1}$, the track $(q_0, \mathsf{true})$ or $(q_0, \mathsf{false})$ is added and its occurrence is unique since $q_0$ there is no edge entering $q_0$ in $\mathcal{A}$. Furthermore, a track $(q, \mathsf{false})$ 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 $\mathsf{merged}(\cdot)$. Thus, no state $q \in Q$ appears twice in $(s_{k+1}, f_{k+1})$. □

**Theorem A.4 (See Theorem III.5).** *Let $\mathcal{T}$ be a strongly connected KS, $\mathcal{A}$ an NFA encoding a co-safety property $\varphi$ and $\mathcal{G}$ the weighted KS defined as above. Then, $\mathbb{LF}^{\max}_{\mathcal{T}}(\varphi) = \mathbb{MP}^{\max}_{\mathcal{G}}(\Phi)$ and $\mathbb{LF}^{\min}_{\mathcal{T}}(\varphi) = \mathbb{MP}^{\min}_{\mathcal{G}}(\Phi)$.*

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

(a) For each infinite path $\pi$ in $\mathcal{T}$ starting in $s_0$ there exists an infinite path $\pi'$ in $\mathcal{G}$ with $\pi' \models \Phi$ and $\mathit{lrf}_{\varphi}(\pi) = \mathit{mp}(\pi')$.
(b) For each initial infinite path $\pi'$ in $\mathcal{G}$ with $\pi' \models \Phi$ there is an infinite path $\pi$ in $\mathcal{T}$ starting in $s_0$ such that $\mathit{lrf}_{\varphi}(\pi) = \mathit{mp}(\pi')$

*Proof of statement (a).:* Consider any infinite path $\pi = (s_k)_{k \geqslant 0}$ of $\mathcal{T}$ that starts at $s_0$. We will construct a uniquely defined path $(s_k, f_k)_{\geqslant 0}$ in $\mathcal{G}$ satisfying $\Phi$ and whose mean payoff is exactly $\mathit{lrf}_{\varphi}(\pi)$.

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 $b \in \{\mathsf{true}, \mathsf{false}\}$ that is introduced at each step, and in the choice of a successor for each $(q, \mathsf{true})$. In fact, the rest of the components are uniquely determined by the construction.

We are going to construct a path $(s_k, f'_k)_{k \geqslant 0}$ with an additional information: at each step $k$, tracks of the form $(q, \mathsf{true})$ are replaced by $(q, \mathsf{true}, \rho)$ where $\rho$ is a path witnessing that the trace $\mathsf{L}(s_k)\mathsf{L}(s_{k+1})\dots$ is accepted by $\mathcal{A}$ from state $q$. The final path $(s_k, f_k)_{k \geqslant 0}$ is then obtained by removing this additional component.

We define $f'_0$ by choosing $(q_0, \mathsf{true}, \rho)$ if and only if $\pi, 0 \models \phi$, where $\rho$ is some witness accepting path of $\mathcal{A}$ 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 $(q_0, \mathsf{true}, \rho)$ if $\pi, k \models \phi$, with $\rho$ a witness path, and add $(q_0, \mathsf{false})$ otherwise.
- For any $0 \leqslant i \leqslant 2\ell$ with $f'_{k-1}(i) = (q, \mathsf{true}, \rho)$, 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 $\rho = qq'\rho'$ where $q, q'$ are first two states of $\rho$ and $\rho'$ is the rest of the path. We let $f'_k(i) = (q', \mathsf{true}, q'\rho')$.

This sequence thus constructed never produces inconsistent states. This is because all tracks are chosen according to the satisfaction of the property $\varphi$ at given position with given witness accepting paths in $\mathcal{A}$. Now, the mean payoff of $(s_k, f_k)_{k \geqslant 0}$ is obtained as the average of the number of positions in which a track $(q_0, \mathsf{true})$ is created. This is the case exactly when $\varphi$ holds from a given position. Thus, the mean payoff of $(s_k, f_k)_{k \geqslant 0}$ is equal to $\mathit{lrf}_{\varphi}(\pi)$.

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

*Proof of statement (b).:* Conversely, consider a path $(s_k, f_k)_{\geqslant 0}$ of $\mathcal{G}$ satisfying $\Phi$. Let $\pi = (s_k)_{k \geqslant 0}$ be the corresponding path in $\mathcal{T}$. We will prove that for all positions $k$ such that $fs_k$ contains a track with $(q_0, \mathsf{true})$, $\pi, k \models \phi$, and for all other positions $k$, $\pi, k \not\models \phi$. This shows that $\mathit{lrf}_{\varphi}(\pi)$ is equal to the mean payoff of the path $(s_k, f_k)_{k \geqslant 0}$.

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

- Case $i = 0$. Consider $k \geqslant 0$ such that $f_k(0) = (q, \mathsf{true})$. We claim that there exists $k' > k$ and states $q_l \in Q$ for all $k \leqslant l \leqslant k'$ such that $f_l(i) = (q_l, \mathsf{true})$, $q_k = q$ and $q_{k'} \in F$, and $q_k q_{k+1} \dots q_{k'}$ is a path of $\mathcal{A}$. In fact, since $(s_k, f_k)_{k \geqslant 0}$ satisfies $\Phi$, the track $i$ cannot be of the form $(q', \mathsf{true})$ indefinitely from position $k$. Moreover, it can never be of the form $(\mathsf{merged}, j)$ since this would imply $j < 0$. It cannot be either of the form $(q, \mathsf{false})$ before visiting some $(q_f, \mathsf{true})$ 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 $q_k q_{k+1} \dots q_{k'}$ is a path of $\mathcal{A}$ by construction of $\mathcal{G}$.
- Consider $i > 0$ and $k \geqslant 0$ such that $f_k(i) = (q, \mathsf{true})$. We claim that one of the two conditions hold:

1) there exists $k' > k$ and states $q_l \in Q$ for all $k \leqslant l \leqslant k'$ such that $f_l(i) = (q_l, \mathsf{true})$, $q_k = q$ and $q_{k'} \in F$, and $q_k q_{k+1} \dots q_{k'}$ is a path of $\mathcal{A}$.
2) or, there exists $k' > k$ and states $q_l \in Q$ for all $k \leqslant l \leqslant k'-1$ such that $f_l(i) = (q_l, \mathsf{true})$, $q_k = q$ and $q_l q_{l+1} \dots q_{k'-1}$ is a path of $\mathcal{A}$ and $f_{k'}(i) = (\mathsf{merged}, j)$ 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 $(\mathsf{false}_i \vee \mathsf{merged}_i \vee F_i \vee \emptyset_i)$. Consider the least such position $k'$. By construction, we cannot have $(s_{k'}, f_{k'}) \models \mathsf{false}_i$ or $(s_{k'}, f_{k'}) \models \emptyset_i$ since all tracks of the form $Q \times \{\mathsf{true}\}$ are stopped by some $F \times \{\mathsf{true}\}$ or $\mathsf{merged} \times \mathbb{N}$. The former does not hold by hypothesis so there must exist $j < i$ such that $f_{k'}(i) = (\mathsf{merged}, j)$. By construction, there exists $q, q' \in Q$ such that $f_{k'-1}(i) = (q, \mathsf{true})$ and $f_{k'}(j) = (q, \mathsf{true})$ 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 $L(s_{k'})L(s_{k'+1})\dots$. Now if $q_k q_{k+1} \dots q_{k'-1}$ denotes the path of $\mathcal{A}$ defined by the track $i$ from positions $k$ to $k'-1$, then $q_k q_{k+1} \dots q_{k'-1} \cdot \rho$ is an accepting path of $\mathcal{A}$ on a prefix of $L(\pi_{[k\dots]})$.

□

*Computation of the extremal values*

Let us explain the exponential-time algorithm of Corollary III.6 for computing $\mathbb{LF}^{\max}_{\mathcal{T},s}(\varphi)$ and $\mathbb{LF}^{\min}_{\mathcal{T},s}(\varphi)$.

We start by building $\mathcal{G}$ which has size $|S|(3\ell+1)^{2\ell+1}$, which is linear in the size of $\mathcal{T}$ and exponential in that of $\mathcal{A}$. We then compute the strongly connected components of $\mathcal{G}$ 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 $\mathcal{G}$. 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 $\mathcal{T}$, an NFA $\mathcal{A}$ and a rational threshold $\vartheta$, check whether $\mathcal{T}$ has an infinite path $\pi$ with $lrf_{\mathcal{A}}(\pi) \geqslant \vartheta$" 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 $\mathcal{T}$, co-safety property $\varphi$ described by NFA $\mathcal{A}$, and a rational number $\vartheta$, a comparison operator $\trianglerighteq \in \{\geqslant, >, <, \leqslant\}$ one can check in polynomial space whether $\mathbb{LF}^{\max}_{\mathcal{T}}(\varphi) \trianglerighteq \vartheta$.*

*Proof.* As PSPACE equals NPSPACE (Savitch's theorem) it suffices to provide a non-deterministic polynomially space-bounded procedure to check whether $\mathbb{MP}^{\max}_{\mathcal{G}}(\Phi) \trianglerighteq \vartheta$. The idea is to use the weighted KS $\mathcal{G}$, without constructing $\mathcal{G}$ explicitly. To simplify the following argument we concentrate on the case $\trianglerighteq = \geqslant$. The algorithm for strict lower bounds and upper bounds are similar.

The algorithm starts by guessing a state $s_{\mathcal{G}}$ in $S_{\mathcal{G}}$ and checks whether $s_{\mathcal{G}}$ 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 $2(2\ell+1)|S_{\mathcal{G}}|$ containing $s_{\mathcal{G}}$ which satisfies each conjunct of $\Phi$,
(2) there exists a simple cycle $\xi_{\mathrm{MP}}$ containing $s_{\mathcal{G}}$ with mean payoff of at least $\vartheta$.

The first check can be done in polynomial space by guessing the path and using a counter up to $2\ell|S_{\mathcal{G}}|$, 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 $|S_{\mathcal{G}}|$ as all weights are 0 or 1.

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

$$avg(\pi_{[0\dots K]}) \quad = \quad \frac{1}{K} \cdot \sum_{i=0}^{K-1} wgt(\pi_{[i]}).$$

If $K \geqslant a+2b$, then there exists a unique integer $i \geqslant 0$ and some integer $r$ with $0 \leqslant r < a+2^{i+1}b$ such that:

$$\begin{array}{rcl} K & = & a+2^1b+a+2^2b+\ldots a+2^ib+r \\ & = & ia+b(2^{i+1}-1)+r \end{array}$$

If $r < a+b$, then:

$$avg(\pi_{[0\ldots K]}) \;\geqslant\; \frac{\vartheta(2^{i+1}-1)}{2^{i+1}-1+(i+1)a+b}$$

If $r \geqslant a+b$, there exists $m < 2^{i+1}$ such that:

$$avg(\pi_{[0\ldots K]}) \;\geqslant\; \frac{\vartheta(m+2^{i+1}-1)}{m+2^{i+1}-1+ia}$$

Then, for all $\varepsilon > 0$, there exists $K_\varepsilon$ such that for all $K \geqslant K_\varepsilon$, $avg(\pi_{[0\ldots K]}) \geqslant \vartheta - \varepsilon$. This yields:

$$mp(\pi) \;=\; \liminf_{K\to\infty} avg(\pi_{[0\ldots K]}) \;\geqslant\; \vartheta$$

As the mean payoff of $\rho$ coincides with the mean payoff of each of its suffixes, we get:

$$mp(\rho) \;=\; mp(\pi) \;\geqslant\; \vartheta$$

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 $\mathcal{G}$ satisfying $\Phi$, starting in a state of $I(s_0)$ and with $mp(\rho) \geqslant \vartheta$. The task is to show that the sketched nondeterministic algorithm has a computation returning the answer "yes". Pick some state $s_\mathcal{G}$ that occurs infinitely in $\rho$. Then, $s_\mathcal{G}$ is reachable from $I(s_0)$. Let us write $\Phi_i$ for the Büchi condition $\Box\Diamond(\mathsf{false}_i \vee \mathsf{merged}_i \vee F_i \vee \emptyset_i)$. Then, $\Phi = \bigwedge_{i=0}^{2\ell}\Phi_i$ and $\rho$ has a fragment that constitutes a cycle in $\mathcal{G}$ containing state $s_\mathcal{G}$ and satisfying all conjuncts $\Phi_i$ of $\Phi$. We now sketch how to construct new a 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 $s_\mathcal{G}$ to some state $t_i$ with $t_i \models \Phi_i$, and a path from $t_i$ to $s_\mathcal{G}$. A cycle $\xi_i$ of length $\leqslant 2|S_\mathcal{G}|$ containing $s_\mathcal{G}$ and $t_i$ can then be constructed by concatenating simple paths from $s_\mathcal{G}$ to $t_i$ and from $t_i$ to $s_\mathcal{G}$ in $\mathcal{G}$. 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 $\mathcal{T}$, an NFA $\mathcal{A}$ and a rational threshold $\vartheta$, decide whether $\mathbb{LF}^{\max}_{\mathcal{T}}(\varphi) \geqslant \vartheta$" 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 $\mathcal{D}_1,\ldots,\mathcal{D}_k$ 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].

To provide a polynomial reduction from the intersection problem for DFA, we suppose we are given DFA $\mathcal{D}_1,\ldots,\mathcal{D}_k$ over some alphabet $\Sigma$. W.l.o.g. we may assume that $k \geqslant 2$ and that the empty word is not included in any of the languages $\mathcal{L}(\mathcal{D}_i)$. Let $Q_i$ be the state space of $\mathcal{D}_i$, $\ell_i = |Q_i|$ and $\ell = \ell_1 \cdot \ldots \ell_k$. Then, $\mathcal{L}(\mathcal{D}_1)\cap\ldots\cap\mathcal{L}(\mathcal{D}_k)$ is nonempty if and only if there is a word $w \in \Sigma^*$ of length at most $\ell$ such that $w \in \mathcal{L}(\mathcal{D}_i)$ for $i = 1,\ldots,k$.

Let $\$_1,\ldots,\$_k,\#$ be pairwise distinct fresh letters (not contained in $\Sigma$), and let $\Gamma = \Sigma \cup \{\$_1,\ldots,\$_k,\#\}$.

Given a finite word $w = \sigma_1\sigma_2\ldots\sigma_n \in \Sigma^+$, let $\hat{w}$ denote the word over $\Sigma\cup\{\#\}$ that arises from $w$ by inserting $(k-1)$-times the symbol # after each letter $\sigma_j$. That is,

$$\hat{w} \;=\; \sigma_1\#^{k-1}\sigma_2\#^{k-1}\ldots\sigma_n\#^{k-1}$$

For $i = 1,\ldots,k$, one can easily construct in time $\mathcal{O}(k^2 + k\cdot size(\mathcal{D}_i))$ a new DFA $\mathcal{B}_i$ over the alphabet $\Gamma$ such that:

$$\mathcal{L}(\mathcal{B}_i) = \left\{ \$_i^j\,\$_{i+1}^{k-1}\ldots\$_k^{k-1}\,\hat{w} : w \in \mathcal{L}(\mathcal{D}_i), 1 \leqslant j < k \right\}$$

Furthermore, we can construct in time linear in the sizes of $\mathcal{B}_1,\ldots,\mathcal{B}_k$ an NFA $\mathcal{A}$ over the alphabet $\Gamma$ with:

$$\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{B}_1)\cup\ldots\cup\mathcal{L}(\mathcal{B}_k)\cup\{\#^i : i \geqslant 1\}$$

Note that $\mathcal{A}$ 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 $\mathcal{T}$ 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 $\{\gamma\}$ with $\gamma$, where $\gamma$ ranges over all symbols of the alphabet $\Gamma$. The labeling function of $\mathcal{T}$ is then given by:

$$L(s_{i,j}) = \$_i,\; L(t_j) = \# \text{ and } L(u_\sigma) = \sigma.$$

The transition relation of $\mathcal{T}$ is depicted in Figure 4.

Thus, the words generated by $\mathcal{T}$ are the substrings of the infinite words $y_1y_2y_3\ldots$ where each word $y_i$ has the form

$$\$_1^{k-1}\$_2^{k-1}\ldots\$_k^{k-1}\hat{w}$$

for some $w \in \Sigma^+$. Let

$$\vartheta = \frac{k(k-1)+(k-1)\ell}{k(k-1)+k\ell}$$

Clearly, $\mathcal{T},\mathcal{A},\vartheta$ can be constructed in time polynomial in the size of the DFA $\mathcal{D}_1,\ldots,\mathcal{D}_k$. It remains to show that $\mathcal{T}$ has an infinite path $\pi$ with long-run frequency

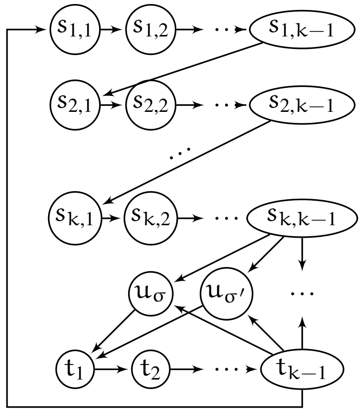


Fig. 4. The Kripke structure $\mathcal{T}$ in the reduction of Theorem III.7.

$lrf_{\mathcal{A}}(\pi)$ at least $\vartheta$ if and only if the intersection language of the $\mathcal{D}_i$'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 $\vartheta$ if and only if the intersection language of the $\mathcal{D}_i$'s is nonempty.

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

$$\begin{array}{ll} s_{i,1} \to s_{i,2} \to \ldots \to s_{i,k-1} & \text{for } i = 1,\ldots,k \\ s_{i,k-1} \to s_{i+1,1} & \text{for } i = 1,\ldots,k-1 \\ s_{k,k-1} \to u_\sigma \to t_1 & \text{for } \sigma \in \Sigma \\ t_1 \to t_2 \to \ldots \to t_{k-1} & \\ t_{k-1} \to s_{1,1} \text{ and } t_{k-1} \to u_\sigma & \text{for } \sigma \in \Sigma \end{array}$$

Suppose first that there is some word $w \in \Sigma^*$ accepted by each of the DFSs $\mathcal{D}_1, \ldots, \mathcal{D}_k$. As stated before, we then can safely assume that $|w| \leqslant \ell$. $\mathcal{T}$ has a cycle $\xi$ generating the word $v = \$_1^{k-1}\$_2^{k-1}\ldots\$_k^{k-1}\hat{w}$. We then have:

$$|v| \;=\; k(k-1) + k|w|$$

The word $v$ 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 $\$_i^j\$_{i+1}^{k-1}\ldots\$_k^{k-1}\hat{w}$ of $v$ are accepted by $\mathcal{B}_i$, and therefore by $\mathcal{A}$ (for $j = 1,\ldots,k-1$).
- $\hat{w}$ 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 infinity is:

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

where $f : \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$ is the following function:

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

Function $f$ is monotonically decreasing.[1] Therefore, $f(|w|) \geqslant f(\ell)$ As $f(\ell) = \vartheta$, we conclude that $\mathcal{T}$ has an infinite path with long-run frequency at least $\vartheta$.

We assume now that $\mathcal{T}$ has an infinite path $\pi$ with long-run frequency at least $\vartheta$. We first observe that $\pi$ must visit $s_{1,1}$ infinitely often as otherwise $\pi$ would have an infinite suffix consisting of $t$- and $u$-states, in which case the long-run frequency would be smaller or equal $(k-1)/k$, and therefore strictly smaller than $\vartheta$. 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 $s_{1,1}s_{1,2}\ldots s_{1,k-1}\ldots s_{k,1}s_{k,2}\ldots s_{k,k-1}\varpi s_{1,1}$ where $\varpi$ 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 $\varpi$, in which case the number of $t$-states in $\varpi$ 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 $\vartheta$. 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 $\vartheta$. $\square$

*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 PSPACE-hard,*
(c) $lrf_{\mathcal{A}}(\pi) = 0$ *is NP-hard,*
(d) $lrf_{\mathcal{A}}(\pi) = 1$ *is 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 "$\Longleftarrow$" we pick a finite word $w$ over $2^{\mathsf{AP}}$ that is accepted by $\mathcal{T} \otimes \mathcal{A}$. As $Q_0 \cap F = \varnothing$, $w$ 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 a pick finite path $t_0 t_1 \ldots t_m$ from $s_n = t_0$ to $s_0 = t_m$, 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 $lrf_{\mathcal{A}}(\pi) > 0$.

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

- For the implication "$\Longrightarrow$", we suppose that we are given an infinite path $\pi$ with $lrf_{\mathcal{A}}(\pi) > 0$. Then, there is a pair $(i, j)$ of integers with $i < j$ such that the word induced by the path fragment $\pi_{[i\dots j]}$ is accepted by $\mathcal{A}$. This path fragment can be lifted to an accepting run in $\mathcal{T} \otimes \mathcal{A}$. Thus, the language of $\mathcal{T} \otimes \mathcal{A}$ is nonempty.

We prove statement (b) via a polynomial reduction from the universality of finite automata, which is known to be PSPACE-complete [33], to the problem that takes as input a KS $\mathcal{T}$ and an NFA $\mathcal{A}$ and asks whether $lrf_{\mathcal{A}}(\pi) = 1$ for all infinite paths $\pi$ in $\mathcal{T}$. Because PSPACE is closed under complement, this implies that checking the existence of an infinite path $\pi$ such that $lrf_{\mathcal{A}}(\pi) < 1$ is PSPACE-hard. For the reduction from the universality problem for NFA we may restrict to NFA over the alphabet $\Sigma = 2^{\mathsf{AP}}$ for some fixed set $\mathsf{AP}$ of atomic propositions. As checking whether the empty word is accepted by an NFA is trivial, we may assume some preprocessing of the given NFA $\mathcal{A}$ that transforms $\mathcal{A}$ into an NFA that accepts the same non-empty words and does not accept the empty word. (So, instead of checking universality of the new NFA, the task is to check whether it accepts all non-empty words.) Given such an NFA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ with $Q_0 \cap F = \varnothing$, we define a KS $\mathcal{T}$ with state space $S = \{s_\sigma : \sigma \in \Sigma\}$, the obvious labeling function $\mathsf{L}(s_\sigma) = \sigma$, and with transitions between all pairs of states. Thus, all words on $\Sigma$ can be generated by $\mathcal{T}$. Consider any NFA $\mathcal{A}$ over $\Sigma$, and some infinite path $\pi$ of $\mathcal{T}$. If $lrf_{\mathcal{A}}(\pi) < 1$, then there exists a position $i \in \mathbb{N}$ such that none of the words $\pi_{[i\dots n]}$ is not accepted by $\mathcal{A}$. So, $\mathcal{A}$ does not accept all non-empty words. Conversely, if $\mathcal{A}$ does not accept all non-empty words, consider a word $w \in \Sigma^+$ rejected by $\mathcal{A}$. Then, $lrf_{\mathcal{A}}(w^\omega) \leqslant \frac{|w|-1}{|w|} < 1$.

For the proof of statement (c), we describe a polynomial reduction from 3SAT. Let $\psi = c_1 \vee \ldots \vee c_m$ be a 3CNF formula with $m$ clauses using $n$ Boolean variables, say $x_1, \ldots, x_n$. Let $\mathcal{T}$ 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 $c_1$. 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 $c_{m+1} = \perp$ whose only outgoing transition goes to $c_1$. The automaton $\mathcal{A}$ 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 $\mathcal{T}$ such that $lrf_{\mathcal{A}}(\pi) = 0$. If $\psi$ is not satisfiable, for all $\pi$ in $\mathcal{T}$, at each cycle, some $x_i$ and $\neg x_i$ must be read. Thus, $lrf_{\mathcal{A}}(\pi) \geqslant \frac{1}{n+m+1}$.

The proof of statement (d) is also a reduction from 3SAT and uses the same KS $\mathcal{T}$ as in the previous construction. Consider 3CNF formula $\psi = c_1 \vee \ldots \vee c_m$ over variables $x_1, \ldots, x_n$, and let us write $c_i = l_1^i \vee l_2^i \vee l_3^i$ where $l_k^i$ are literals. Let $L$ denote the set of all literals, and $C = \{c_1, \ldots, c_m\}$. We define $\mathcal{A}$ as follows. From the initial state, one goes to an accepting sink state by reading any letter of $L \cup \{\perp\}$. By reading $c_i \in C$, one goes to $s_{c_i}$ from which the automaton accepts when $\perp$ 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 $l_k^i$ 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 $w = c_1 l_{i_1}^1 c_2 l_{i_2}^2 \ldots c_m l_{i_m}^m \perp$ where $l_{i_k}^k \in c_k$ for each $k$, and such that for all $k, k'$, $l_{i_k}^k \neq \neg l_{i_{k'}}^{k'}$. Consider the word $\pi = w^\omega$. The property $\mathcal{A}$ is satisfied from all positions. This is trivial from positions containing $X \cup \{\perp\}$. 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, $lrf_{\mathcal{A}}(\pi) = 1$.

Assume now that $\psi$ is not satisfiable. Consider any word $\pi$ of $\mathcal{T}$ which can be written as $\pi = w_1 \perp w_2 \perp \ldots$ where each $w_i$ has the form $c_1 l_1 c_2 l_2 \ldots c_m l_m$ with $l_k \in c_k$ for each $k$. Since $\psi$ is not satisfiable, each $w_i$ must contain $l_j$ and $l_k$ with $l_j = \neg l_k$. That is, at least once every $m+n+1$ positions, the property is violated, Thus, $lrf_{\mathcal{A}}(\pi) < 1$. $\square$

### 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 $\mathcal{E}$ as a pair $(E, \mathfrak{A})$ where $E$ is a set of states and $\mathfrak{A} : E \to Act$, namely $E = \{s \in S : \exists \alpha \in Act. (s, \alpha) \in \mathcal{E}\}$ and $\mathfrak{A}(s) = \{\alpha \in Act : (s, \alpha) \in \mathcal{E}\}$. With this representation is mind and identifying $\mathcal{E}$ with $E$, we sometimes use notations like $s \in \mathcal{E}$ or $T \cap \mathcal{E}$ for $s \in S$ and $T \subseteq S$.

If $\mathfrak{S}$ is a scheduler and $\pi = s_0 \alpha_0 s_1 \alpha_1 \ldots$ a path then $\pi$ is said to be a $\mathfrak{S}$-path if $\mathfrak{S}(s_0 \alpha_0 \ldots \alpha_{n-1} s_n)(\alpha_n) > 0$ for all $n \geqslant 0$. 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 $\mathfrak{S}$-paths constitutes an end component. Here, the limit $Lim(\pi)$ 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 $\mathbb{LP}^{\max}_{\mathcal{M},s}(\varphi)$ and $\mathbb{LP}^{\min}_{\mathcal{M},s}(\varphi)$ are computable in polynomial-time if $\varphi$ is a condition of one the following types:*

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

*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 $\mathbb{LP}^{\max}_{\mathcal{M},s}(\varphi)$. The argument for $\mathbb{LP}^{\min}_{\mathcal{M},s}(\varphi)$ 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 $\mathfrak{S}$ that maximizes the probability for $\varphi$ from every visited state in the following sense:

$$\Pr^{\mathfrak{S}\uparrow\pi_{[0\ldots i]}}_{\mathcal{M},\pi_{[i]}}(\varphi) \;=\; \Pr^{\max}_{\mathcal{M},\pi_{[i]}}(\varphi)$$

for each infinite $\mathfrak{S}$-path $\pi$ and each position $i \in \mathbb{N}$. For reachability, invariances, Büchi and co-Büchi conditions, we may even suppose that $\mathfrak{S}$ is an MD-scheduler with a single BSCC $\mathcal{B}$.

If $\varphi$ is a reachability, generalized Rabin or Streett condition then $\Pr^{\max}_{\mathcal{M},s}(\varphi) = \Pr^{\max}_{\mathcal{M},t}(\varphi)$ for all states $s, t$ in $\mathcal{M}$. Moreover, this value is either 0 or 1. But then $\mathfrak{S}$ 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 $\max_{s\in S} \Pr^{\max}_{\mathcal{M},s}(\varphi) = 0$, in which case $\mathbb{LP}^{\max}_{\mathcal{M},s}(\varphi) = 0$ for all states $s$, or the unique BSCC $\mathcal{B}$ of $\mathfrak{S}$ consists of b-states. In the latter case, $\Pr^{\mathfrak{S}}_{\mathcal{M},s}(\Box b) = \Pr^{\max}_{\mathcal{M},s}(\Box b) = 1$ for all states $s$ in $\mathcal{B}$. Let now $\mathfrak{T}$ be the following MD-scheduler:

- From the states not in $\mathcal{E}$, $\mathfrak{T}$ mimics an MD-scheduler maximizing the probability to reach $\mathcal{B}$ (which is 1 as $\mathcal{M}$ is strongly connected).
- For the state inside $\mathcal{B}$, $\mathfrak{T}$ behaves as $\mathfrak{S}$.

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

### *Qualitative lrp-problems*

Recall that the task of the qualitative lrp-problems is to decide the existence of a scheduler $\mathfrak{S}$ such that $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi)$ 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:*

$$\begin{array}{lcl} \mathbb{LP}^{\max}_{\mathcal{M}}(a\,\mathrm{U}\,b) = 1 & \textit{iff} & \exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) = 1 \\ \mathbb{LP}^{\min}_{\mathcal{M}}(a\,\mathrm{U}\,b) = 0 & \textit{iff} & \exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) = 0 \end{array}$$

*Proof.* Let $\varphi = a\,\mathrm{U}\,b$ 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) $\exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) > 0$ iff $B \neq \varnothing$
(b) $\exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) = 1$ iff $\mathcal{M}$ has a $(A \cup B)$-EC $\mathcal{E}$ with $\mathcal{E} \cap B \neq \varnothing$ or $\mathcal{M}$ has an $A$-EC $\mathcal{E}$ with $\Pr^{\max}_{\mathcal{E}}(\varphi) = 1$ (or both).
(c) $\exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) < 1$ iff $\mathcal{M}$ has an $A$-EC or $C \neq \varnothing$
(d) $\exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) = 0$ 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) $\mathbb{LP}^{\max}_{\mathcal{M}}(\varphi) = 1$
(ii) $\exists\mathfrak{S}.\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi) = 1$
(iii) $\mathcal{M}$ has a $(A \cup B)$-EC $\mathcal{E}$ with $\mathcal{E} \cap B \neq \varnothing$ or $\mathcal{M}$ has an $A$-EC $\mathcal{E}$ with $\Pr^{\max}_{\mathcal{E}}(\varphi) = 1$ (or both).

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

"(i) $\Longrightarrow$ (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 $\Pr^{\min}_{\mathcal{M},s}(\Diamond^{\leqslant n} C) > 0$ for every state $s$. As $\Pr^{\max}_{\mathcal{M},t}(a\,\mathrm{U}\,b) = 0$ for each $C$-state $t$, this yields

$$\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a\,\mathrm{U}\,b) \;\leqslant\; \frac{n-1}{n} \;<\; 1$$

for each scheduler $\mathfrak{S}$. Hence, $\mathbb{LP}^{\max}_{\mathcal{M}}(a\,\mathrm{U}\,b) \leqslant (n-1)/n < 1$.

The implication "(ii) $\Longrightarrow$ (i)" is trivial.

The proofs for statements (c) and (d) and the equivalence of $\mathbb{LP}^{\min}_{\mathcal{M}}(a\,\mathrm{U}\,b) = 0$ and the existene of a scheduler $\mathfrak{S}$ with $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a\,\mathrm{U}\,b) = 0$ are similar and omitted here. $\square$

### *Duality of minimal and maximal long-run probabilities*

The following lemma essentially shows that minimizing the long-run probability for $a\,\mathrm{U}\,b$ is dual to the task to maximize the long-run probability for $A\,\mathrm{U}\,C$. An exception is the case where $\mathcal{M}$ has $A$-ECs, in which case the minimal lrp-problem can be answered directly.

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

*Proof.* Suppose first that $\mathcal{M}$ has an $A$-EC $\mathcal{E}$. For each state $t$ in $\mathcal{E}$, let $\mathfrak{A}(t) = \{\alpha \in Act : (s, \alpha) \in \mathcal{E}\}$. Consider an MD-scheduler $\mathfrak{S}$ maximizing the probability to reach $\mathcal{E}$ from every state $s$ outside $\mathcal{E}$ and which selects only actions in $\mathfrak{A}(t)$ for every state $t$ in $\mathcal{E}$. Then, $\Pr^{\mathfrak{S}}_{\mathcal{M},s}(\Diamond\Box A) = 1$ for all states $s$, and therefore $\Pr^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, B) = 0$ (as $A$ and $B$ are disjoint). But then $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) = \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, B) = 0$ for each state $s$. This yields $\mathbb{LP}^{\min}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 0$.

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

$$\Pr^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) = \Pr^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, B) = 1 - \Pr^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, C)$$

This yields $\mathbb{LP}^{\min}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 1 - \mathbb{LP}^{\max}_{\mathcal{M}}(A \,\mathrm{U}\, C)$. ☐

*Finite-memory schedulers for $a \,\mathrm{U}\, b$*

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

As before, we suppose that $\mathcal{M}$ is a strongly connected MDP with state space $S$. Furthermore, we may safely assume that $C \neq \varnothing$ as otherwise all states either belong to $A$ or $B$, in which case either $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 0$ if $B = \varnothing$ or, if $B \neq \varnothing$, then $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) = \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 1$ for any MD-scheduler $\mathfrak{S}$ that maximizes the probability for reaching $B$.

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

$$\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) \;\geqslant\; \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) - \varepsilon$$

*Proof.* If $\mathfrak{S}$ is a scheduler then briefly we write $p^{\mathfrak{S}}_s$ instead of $\Pr^{\mathfrak{S}}_{\mathcal{M},s}(A \,\mathrm{U}\, B)$.

By Fatou's lemma, we have:

$$\begin{aligned}\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) \;&=\; \mathbb{E}^{\mathfrak{T}}_{\mathcal{M},s}\left(\liminf_{n\to\infty} \frac{1}{n+1}\sum_{i=0}^{n} p_{\pi[i]}^{\mathfrak{T}\uparrow\pi[0\ldots i]}\right)\\ &\leqslant\; \liminf_{n\to\infty}\, \mathbb{E}^{\mathfrak{T}}_{\mathcal{M},s}\left(\frac{1}{n+1}\sum_{i=0}^{n} p_{\pi[i]}^{\mathfrak{T}\uparrow\pi[0\ldots i]}\right)\end{aligned}$$

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

$$\mathbb{E}^{\mathfrak{T}}_{\mathcal{M},s}\left(\frac{1}{k+1}\sum_{i=0}^{k} p_{\pi[i]}^{\mathfrak{T}\uparrow\pi[0\ldots i]}\right) \;\geqslant\; \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) - \frac{\varepsilon}{2}$$

Let $\mathfrak{U}_s$ 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 $a \,\mathrm{U}\, b$. As soon as a state in $B \cup C$ has been reached, scheduler $\mathfrak{U}_s$ operates in the second mode, in which it memorylessly minimizes the expected number of steps until reaching $s$. Let $f_{t,s} = \mathbb{E}^{\min}_{\mathcal{M},t}$("steps until $s$") denote the expected number of steps this scheduler $\mathfrak{U}_s$ needs to reach $s$ in the second mode starting from state $t$. We then define $f_s = \max_{t\in S} f_{t,s}$ and $f = \max_{s\in S} f_s$.

We now construct an FM-scheduler $\mathfrak{S}$ satisfying the claim of the lemma. First, choose a natural number $k$ with $k \geqslant k_0$ and $k+1 > \frac{2f_s}{\varepsilon}$. The behavior of scheduler $\mathfrak{S}$ is as follows. In its first mode, it starts in $s$ and behaves like $\mathfrak{T}$ in the first $k$ steps. Then, it switches to the second mode and behaves like $\mathfrak{U}_s$ until it reaches $s$ (in the second mode of $\mathfrak{U}_s$). Afterwards, it switches back to the first mode.

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

$$\frac{1}{k+1}\cdot\sum_{i=0}^{k} p_{\pi[i]}^{\mathfrak{T}\uparrow\pi[0\ldots i]} \quad\leqslant\quad \frac{1}{k+1}\cdot\sum_{i=0}^{k} p_{\pi[i]}^{\mathfrak{S}\uparrow\pi[0\ldots i]}$$

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

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

$$\begin{aligned}&\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)\\ &\quad\geqslant\quad \frac{p_{\pi[0]}^{\mathfrak{T}\uparrow\pi[0\ldots 0]} + \ldots + p_{\pi[k]}^{\mathfrak{T}\uparrow\pi[0\ldots k]}}{k+1+f_s}\\ &\quad\geqslant\quad \frac{p_{\pi[0]}^{\mathfrak{T}\uparrow\pi[0\ldots 0]} + \ldots + p_{\pi[k]}^{\mathfrak{T}\uparrow\pi[0\ldots k]}}{(k+1)\cdot(1+\varepsilon/2)}\\ &\quad\geqslant\quad \frac{p_{\pi[0]}^{\mathfrak{T}\uparrow\pi[0\ldots 0]} + \ldots + p_{\pi[k]}^{\mathfrak{T}\uparrow\pi[0\ldots k]}}{k+1}\cdot(1-\varepsilon/2)\\ &\quad\geqslant\quad (\mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) - \varepsilon/2)\cdot(1-\varepsilon/2)\end{aligned}$$

by the choice of $k$. Using the fact that $\mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$ is bounded by 1 we obtain:

$$\begin{aligned}&\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)\\ &\quad\geqslant\quad (\mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) - \varepsilon/2)\cdot(1-\varepsilon/2)\\ &\quad\geqslant\quad \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) - \varepsilon\end{aligned}$$

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

*Saturation point*

We now present the proof of Lemma IV.7.

**Lemma A.13 (see Lemma IV.7).** *Suppose $\mathcal{M}$ has no $(A \cup B)$-EC containing at least one $B$-state. Then, for* *each FM-scheduler $\mathfrak{T}$, there is a scheduler $\mathfrak{S} \in \mathrm{FM}(K)$ with $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M}}(a \,\mathrm{U}\, b) \;\geqslant\; \max_{s\in S} \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$.*

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

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

If $\mathcal{B}^{\mathfrak{T}}$ consists of $A$-states then $\mathbb{LP}^{\mathfrak{T}}_{\mathcal{M}}(a \,\mathrm{U}\, b) = 0$ and the claim is trivial as we can deal with any $\mathrm{FM}(K)$-scheduler.

Suppose now that $\mathcal{B}^{\mathfrak{T}}$ contains at least one state in $B \cup C$. As $\mathcal{M}$ has no $(A \cup B)$-EC containing at least one $B$-state and as $\mathcal{B}^{\mathfrak{T}}$ constitutes an EC, $\mathcal{B}^{\mathfrak{T}}$ must contain at least one $C$-state.

We now explain how to modify $\mathfrak{T}$'s decision for generating a scheduler in $\mathrm{FM}(K)$ with the desired property. Our procedure works by induction on the number $k^{\mathfrak{T}}$ of states $\mathfrak{s} = (s,x)$ in $\mathcal{B}^{\mathfrak{T}}$ where $\Pr_{\mathcal{B}^{\mathfrak{T}},\mathfrak{s}}(A \,\mathrm{U}^{\geqslant K}\, D^{\mathfrak{T}}) > 0$. Here, $D^{\mathfrak{T}}$ denotes the set of $A$-states $\mathfrak{t} = (t,y)$ in $\mathcal{B}^{\mathfrak{T}}$ where $\mathfrak{T}(\mathfrak{t})(\alpha) > 0$ for some $\alpha \notin Act^{\max}(t)$.

If $k^{\mathfrak{T}} = 0$ then we can deal with $\mathfrak{S} = \mathfrak{T}$. Suppose now that $k^{\mathfrak{T}} \geqslant 1$. We now show how to transform $\mathfrak{T}$ into a new FM-scheduler $\mathfrak{S}$ with a single BSCC such that $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M}}(a \,\mathrm{U}\, b) \geqslant \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ and $k^{\mathfrak{S}} < k^{\mathfrak{T}}$.

Given states $\mathfrak{s} = (s,x)$ and $\mathfrak{t} = (t,y)$ in $\mathcal{B}^{\mathfrak{T}}$, let $\Pi_{\mathfrak{s},\mathfrak{t}}$ denote the set of of finite $\mathfrak{T}$-paths $\varpi = \mathfrak{s}_1 \alpha_1 \ldots \alpha_{n-1} \mathfrak{s}_n$ such that $n \geqslant K$, $\mathfrak{s}_o = \mathfrak{s}$, $\mathfrak{s}_n = \mathfrak{t}$, $\mathfrak{s}_o, \ldots, \mathfrak{s}_n$ are $A$-states and $\mathfrak{T}(\mathfrak{s}_n)(\alpha) > 0$ for some action $\alpha \notin Act^{\max}(t)$. Let $\Pi_{\mathfrak{s}}$ denote the union of the sets $\Pi_{\mathfrak{s},\mathfrak{t}}$.

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

The definition of FM-scheduler $\mathfrak{S}$ is as follows. Scheduler $\mathfrak{S}$ operates in two phases. Its first phase starts in $\mathfrak{s}$ and uses additional memory cells to keep track of the number of consecutive $A$-states that have been traversed since the last visit of $\mathfrak{s}$. As long as this number is smaller than $K$ or if a $B \cup C$-state has been reached along a path where this number is always smaller than $K$, scheduler $\mathfrak{S}$ just behaves like $\mathfrak{T}$. As soon as this number is $K$, scheduler $\mathfrak{S}$ switches to the second phase and behaves as scheduler $\mathfrak{R}_{N,s}$ (viewed as an FMD-scheduler for $\mathcal{M}$, i.e. it uses the number of consecutive $A$-states in the additional memory cells). Since $K \geqslant N = |A|$, $\mathfrak{R}_{N,s}$ will only choose actions in $Act^{\max}$ until $A$ is left. As soon as $s$ is reached, $\mathfrak{S}$ switches back to the first phase and restarts to mimic $\mathfrak{T}$. For all states that are not reachable from $\mathfrak{s}$ in this way, $\mathfrak{S}$ behaves as $\mathfrak{T}$.

As $\mathfrak{T}$ has a single BSCC, so does $\mathfrak{S}$, and $\mathfrak{s}$ is reached infinitely often almost surely with finite expected return time under $\mathfrak{S}$ and $\mathfrak{T}$.

Let us first observe that we indeed have $k^{\mathfrak{S}} < k^{\mathfrak{T}}$. This is thanks to the fact that $\mathfrak{R}_{N,s}$ maximizes the probability for $a \,\mathrm{U}\, b$ whenever $K$ or more consecutive $A$-states have been visited.

We now show that $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) \geqslant \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$. To simplify the calculations, we present the proof for the case where $\Pi_{\mathfrak{s}}$ is a singleton, say $\Pi_{\mathfrak{s}} = \{\varpi\}$.

The long run probabilities of the two schedulers $\mathfrak{S}$ and $\mathfrak{T}$ can be expressed as follows.

Let $ens^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}$ be the expected return time, i.e. the expected number of steps, from $\mathfrak{s}$ to $\mathfrak{s}$ under $\mathfrak{T}$, and let $ens^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}}$ be the expected time that $\mathfrak{T}$ needs from $\mathfrak{t}$ to $\mathfrak{s}$ if it chooses $\alpha$ in $\mathfrak{t}$ first. Furthermore, let $eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}$ and $eap^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}}$ be the expected accumulated $\Pr^{\mathfrak{T}}(a \,\mathrm{U}\, b)$ that $\mathfrak{T}$ accumulates during these periods. Then:

$$\mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) \;=\; \frac{eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}}{ens^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}}$$

For the scheduler $\mathfrak{S}$ we express $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b)$ as the fraction of expected accumulated probability and expected return time from $\mathfrak{s}$ to $\mathfrak{s}$ as well. Recall that $e_{t,s}$ is the expected time that $\mathfrak{R}_{N,s}$ needs from $t$ to $s$ in its second mode.

For $0 \leqslant i \leqslant K$, let $\wp_{[0\ldots i]}$ denote the probability under $\mathfrak{T}$ for generating the path fragment $\varpi_{[0\ldots i]}$ from state $s$ in mode $x$. So, $\wp_{[0\ldots K]}$ is the probability under $\mathfrak{T}$ for generating $\varpi$ from $\mathfrak{s}$. Likewise, we write $\wp_{[i\ldots N]}$ for the probability under $\mathfrak{T}$ for generating the path fragment $\varpi_{[i\ldots N]}$ from state $\varpi_{[i]}$ in the corresponding mode.

So, the expected time that $\mathfrak{S}$ needs from $\mathfrak{s}$ to $\mathfrak{s}$ is:

$$ens^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}} + \wp_{[0\ldots K]} \cdot p \cdot (e_{t,s} - ens^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}})$$

The accumulated probability on the other hand is at least:

$$\begin{aligned}
& eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}} + \sum_{i=1}^{K} \wp_{[0\ldots i]} \cdot \wp_{[i\ldots K]} \cdot p \cdot (p^{\max}_t - p_{t,\alpha}) \\
& \qquad - \; \wp_{[0\ldots K]} \cdot p \cdot eap^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}} \\
\geqslant \quad & eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}} + K \cdot \wp_{[0\ldots K]} \cdot p \cdot \delta - \wp_{[0\ldots K]} \cdot p \cdot eap^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}}
\end{aligned}$$

With $q = \wp_{[0\ldots K]} \cdot p$, we obtain:

$$\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \,\mathrm{U}\, b) \;\geqslant\; \frac{eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}} + q \cdot (K \cdot \delta - eap^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}})}{ens^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}} + q \cdot (e_{t,s} - ens^{\mathfrak{T}}_{\mathfrak{t},\alpha,\mathfrak{s}})}$$

As $K \cdot \delta \geqslant e$, we get:

$$\begin{aligned} \frac{K \cdot \delta - eap^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}}{e_{t,s} - ens^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}} & \geqslant \frac{K \cdot \delta - ens^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}}{e - ens^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}} \\ & \geqslant \frac{e - ens^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}}{e - ens^{\mathfrak{T}}_{t,\alpha,\mathfrak{s}}} = 1 \\ & \geqslant \frac{eap^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}}{ens^{\mathfrak{T}}_{\mathfrak{s},\mathfrak{s}}} \end{aligned}$$

We conclude that $\mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s}(a \mathrm{U} b) \geqslant \mathbb{LP}^{\mathfrak{T}}_{\mathcal{M},s}(a \mathrm{U} b)$. □

*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 $a \mathrm{U} b$.

**Lemma A.14.** $\mathbb{LP}^{\max}_{\mathcal{M}}(a \mathrm{U} b) = \mathbb{LP}^{\max}_{\mathcal{K}}(a \mathrm{U} b)$

*Proof.* obvious from Lemma IV.7. □

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

$$S_{\mathcal{N}} = C \times \{0\} \cup (A \cup B) \times \{1, \ldots, K, \top\}$$

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

- If $s \in C$ then $P_{\mathcal{N}}((s,0), \alpha, (s',k)) = P(s,\alpha,s')$ if either $s' \in C \wedge k = 0$ or $s' \in A \cup B \wedge k = 1$.
- If $(s,i) \in A \times \{1, \ldots, K{-}1\}$ then:
  $P_{\mathcal{N}}((s,i), \alpha, (s',0)) = P(s,\alpha,s')$ if $s' \in C$
  $P_{\mathcal{N}}((s,i), \alpha, (s',i{+}1)) = P(s,\alpha,s')$ if $s' \in A \cup B$
- If $s \in A$, $k \in \{K, \top\}$ and $\alpha \in Act^{\max}(s)$ then:
  $P_{\mathcal{N}}((s,k), \alpha, (s',0)) = P(s,\alpha,s')$ if $s' \in C$
  $P_{\mathcal{N}}((s,k), \alpha, (s',\top)) = P(s,\alpha,s')$ if $s' \in A \cup B$
- If $s \in B$ and $k \in \{1, \ldots, K, \top\}$ then:
  $P_{\mathcal{N}}((s,k), \alpha, (s',0)) = P(s,\alpha,s')$ if $s' \in C$
  $P_{\mathcal{N}}((s,k), \alpha, (s',1)) = P(s,\alpha,s')$ if $s' \in A \cup B$

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

- $wgt_{\mathcal{N}}(s,k) = k$ if $s \in B$ and $k \in \{1, \ldots, K\}$
- $wgt_{\mathcal{N}}(s,K) = p^{\max}_s \cdot K$ if $s \in A$
- $wgt_{\mathcal{N}}(s,\top) = p^{\max}_s$ if $s \in A \cup B$

Note that for $s \in B$ we have $wgt_{\mathcal{N}}(s,\top) = 1$.

If $s$ is a state of $\mathcal{M}$ then $s_{\mathcal{N}}$ denotes the corresponding state in $\mathcal{N}$, namely $s_{\mathcal{N}} = (s,0)$ for $s \in C$ and $s_{\mathcal{N}} = (s,1)$ if $s \in A \cup B$.

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

**Lemma A.15.** $\mathbb{LP}^{\max}_{\mathcal{K}}(a \mathrm{U} b) = \mathbb{LP}^{\max}_{\mathcal{N}}(a \mathrm{U} b)$ *and* $\mathbb{E}^{\max}_{\mathcal{K}}(\mathbb{MP}) = \mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP})$

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

Hence, the remaining is to prove that $\mathbb{LP}^{\max}_{\mathcal{N}}(a \mathrm{U} b)$ coincides with $\mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP})$.

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

$$\mathbb{E}^{\mathfrak{S}}_{\mathcal{N},s_{\mathcal{N}}}(\mathbb{MP}) \geqslant \mathbb{LP}^{\mathfrak{S}}_{\mathcal{N},s_{\mathcal{N}}}(a \mathrm{U} b)$$

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

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

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

- Symbol $\mathfrak{s}$ is used to denote states in $\mathcal{B}$.
- $\theta_{\mathfrak{s}}$ denotes the long-run frequency (steady-state probability) of state $\mathfrak{s}$ in $\mathcal{B}$.
- $p_{\mathfrak{s}}$ denotes the probability for $a \mathrm{U} b$ in $\mathcal{C}$ (resp. $\mathcal{B}$) from state $\mathfrak{s}$.
- $w_{\mathfrak{s}}$ denotes the weight of state $\mathfrak{s}$ in $\mathcal{C}$, i.e., if $\mathfrak{s} = (s,k,x)$ then $w_{\mathfrak{s}} = wgt_{\mathcal{N}}(s,k)$.

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

$$\begin{aligned} \mathfrak{A} &= \mathcal{B} \cap (A \times \{1, \ldots, K\} \times X) \\ \mathfrak{B} &= \mathcal{B} \cap (B \times \{1, \ldots, K\} \times X) \end{aligned}$$

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

Treating $\theta = (\theta_\mathfrak{s})_{\mathfrak{s}\in S_\mathcal{C}}$ as a row vector and $p = (p_\mathfrak{s})_{\mathfrak{s}\in S_\mathcal{C}}$ and $w = (w_\mathfrak{s})_{\mathfrak{s}\in S_\mathcal{C}}$ as column vectors:

$$\mathbb{E}^{\mathfrak{S}}_{\mathcal{N},s_\mathcal{N}}(\mathbb{MP}) - \mathbb{LP}^{\mathfrak{S}}_{\mathcal{N},s_\mathcal{N}}(a \mathsf{U} b) \quad = \quad \langle \theta, w - p \rangle$$

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

$$\begin{aligned}
\langle \theta, w - p \rangle \quad &= \sum_{\mathfrak{s}\in S_\mathcal{C}} \theta_\mathfrak{s} \cdot (w_\mathfrak{s} - p_\mathfrak{s}) \\
&= \sum_{\mathfrak{s}\in \mathfrak{A}_\top \cup \mathfrak{B}_\top} \theta_\mathfrak{s} \cdot (w_\mathfrak{s} - p_\mathfrak{s}) \\
&\geqslant \sum_{\mathfrak{s}\in \mathfrak{A}\cup\mathfrak{B}} \theta_\mathfrak{s} \cdot (w_\mathfrak{s} - p_\mathfrak{s})
\end{aligned}$$

If $\mathfrak{S}$ leaves $A \times \{\top\}$ with probability 1 after entering, then equality holds, as then $w_\mathfrak{s} = p_\mathfrak{s} = p^{\max}_s$ for each state $\mathfrak{s} = (s,k,x)$ in $\mathcal{B}$ with $s \in A$ and $k = \top$.

We define the matrix

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

as the transition probability matrix of $\mathcal{C}$ restricted to the states in $\mathfrak{A}\cup\mathfrak{B}$ 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 $\mathfrak{A}\cup\mathfrak{B}$.

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

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

Intuitively, the left equation states that we can compute the steady state probabilities for states with index $i$ using only the steady state probabilities for states with index $j$ and the $j-i$ steps transition probability matrix $\mathfrak{P}^{j-i}$. 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 $d \in [0,1]^{\mathfrak{A}\cup\mathfrak{B}}$ with $d_{(s,k,x)} = 1$ if $s \in B$, $d_{(s,K,x)} = p^{\max}_s$, and $d_{(s,k,x)} = 0$ in all other cases. Then:

$$p'[i] \quad \leqslant \quad \sum_{j=0}^{K-i} \mathfrak{P}^j \cdot d$$

where the inequality is understood componentwise. Again, equality holds if $\mathfrak{S}$ always leaves $A \times \{\top\}$ almost surely.

So, we get:

$$\begin{aligned}
&\sum_{\mathfrak{s}\in\mathfrak{A}\cup\mathfrak{B}} \theta_\mathfrak{s} \cdot p_\mathfrak{s} \\
= \quad &\sum_{k=1}^{K} \langle \theta'[k], p'[k] \rangle \quad \leqslant \quad \sum_{k=1}^{K} \Big\langle \theta'[k], \sum_{j=0}^{K-k} \mathfrak{P}^j \cdot d \Big\rangle \\
= \quad &\sum_{k=1}^{K} \sum_{j=0}^{K-k} \theta'[k] \cdot \mathfrak{P}^j \cdot d \\
= \quad &\sum_{k=1}^{K} \sum_{j=0}^{K-k} \theta'[1] \cdot \mathfrak{P}^{j+k-1} \cdot d \\
= \quad &\sum_{k=1}^{K} \sum_{i=K-k}^{K} \theta'[i] \cdot d \quad = \quad \sum_{k=1}^{K} k \cdot q'[k] \cdot d
\end{aligned}$$

But now, we see that the weight structure in $\mathcal{N}$ was just defined such that the last line sums up to $\langle \theta', w' \rangle$. Thus:

$$\sum_{\mathfrak{s}\in\mathfrak{A}\cup\mathfrak{B}} \theta_\mathfrak{s} \cdot p_\mathfrak{s} \quad = \quad \sum_{\mathfrak{s}\in\mathfrak{A}\cup\mathfrak{B}} \theta_\mathfrak{s} \cdot w_\mathfrak{s}$$

We conclude that $\langle \theta, w - p \rangle \geqslant 0$ and $\langle \theta, w - p \rangle = 0$ if $\mathfrak{S}$ leaves $A \times \{\top\}$ almost surely.

This yields $\mathbb{E}^{\mathfrak{S}}_{\mathcal{B}}(\mathbb{MP}) \geqslant \mathbb{LP}^{\mathfrak{S}}_{\mathcal{B}}(a \mathsf{U} b)$ in the general case and $\mathbb{E}^{\mathfrak{S}}_{\mathcal{B}}(\mathbb{MP}) = \mathbb{LP}^{\mathfrak{S}}_{\mathcal{B}}(a \mathsf{U} b)$ if $\mathfrak{S}$ leaves $A \times \{\top\}$ almost surely. □

**Lemma A.17.** $\mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP}) = \mathbb{LP}^{\max}_{\mathcal{N}}(a \mathsf{U} b)$

*Proof.* Thanks to Lemma A.16, it suffices to construct a scheduler $\mathfrak{T}$ for $\mathcal{N}$ such that $\mathbb{LP}^{\mathfrak{T}}_{\mathcal{N},s_\mathcal{N}}(a \mathsf{U} b) = \mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP})$ for all states $s_\mathcal{N}$.

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

Case 1: $\mathcal{B}$ is not contained in $A \times \{\top\}$. Then $\mathfrak{S}$ always leaves $A \times \{\top\}$ with probability 1. The second part of Lemma A.16 yields:

$$\mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP}) \;=\; \mathbb{E}^{\mathfrak{S}}_{\mathcal{N},s_\mathcal{N}}(\mathbb{MP}) \;=\; \mathbb{LP}^{\mathfrak{S}}_{\mathcal{M},s_\mathcal{N}}(a \mathsf{U} b)$$

Hence, we can deal with $\mathfrak{S} = \mathfrak{T}$.

Case 2: $\mathcal{B}$ is contained in $A \times \{\top\}$. In this case, we can construct an infinite-memory scheduler $\mathfrak{T}$ with $\mathbb{LP}^{\mathfrak{T}}_{\mathcal{N},s_\mathcal{N}}(a \mathsf{U} b) = \mathbb{E}^{\mathfrak{S}}_{\mathcal{N},s_\mathcal{N}}(\mathbb{MP})$. The idea is that scheduler $\mathfrak{T}$ stays inside $\mathcal{B}$ for larger and larger number of steps and in between it leaves $\mathcal{B}$ to maximize the probability for $a \mathsf{U} b$. In this way, the weight of states in $\mathcal{B}$ is equal to the probability of $a \mathsf{U} b$ under $\mathfrak{T}$ and the long-run probability then equals this probability. □

**Corollary A.18 (See Lemma IV.8).** $\mathbb{LP}^{\max}_{\mathcal{M}}(a \mathsf{U} b) = \mathbb{E}^{\max}_{\mathcal{K}}(\mathbb{MP})$.

*Proof.* The claim follows by combining the results that have been established so far:

$$\begin{array}{rcll} \mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) & = & \mathbb{LP}^{\max}_{\mathcal{K}}(a \,\mathrm{U}\, b) & \text{(Lemma A.14)} \\ & = & \mathbb{LP}^{\max}_{\mathcal{N}}(a \,\mathrm{U}\, b) & \text{(Lemma A.15)} \\ & = & \mathbb{E}^{\max}_{\mathcal{N}}(\mathbb{MP}) & \text{(Lemma A.17)} \\ & = & \mathbb{E}^{\max}_{\mathcal{K}}(\mathbb{MP}) & \text{(Lemma A.15)} \end{array}$$

□

*NP-hardness*

We now turn to the threshold problem for long-run probabilities and until-properties that takes as input a strongly connected $\mathcal{M}$, atomic propositions $a$, $b$ and a rational value $\vartheta$ and asks whether $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) \geqslant \vartheta$. By the results that have been establishes so far, this problem belongs to EXPTIME. We now provide an NP lower bound.

**Theorem A.19.** *The threshold problem "is $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) \geqslant \vartheta$?" 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 $\mathcal{D}_1, \ldots, \mathcal{D}_k$ over the alphabet $\Sigma = \{0\}$. where $\mathcal{D}_i = (Q_i, \Sigma, \delta_i, q_{0.i}, F_i)$. We simply write $\delta_i(q)$ rather than $\delta_i(q, 0)$. We may suppose the transition functions $\delta_i$ are total and that $Q_i \cap Q_j = \varnothing$ if $i \neq j$. W.l.o.g. we further assume that $|Q_i| \geqslant 2$ for all $i \leqslant k$.

We are going to construct an MDP $\mathcal{M}$ over $\mathsf{AP} = \{a, b\}$ and a rational value $\vartheta$ such that $\mathbb{LP}^{\max}_{\mathcal{M}}(a \,\mathrm{U}\, b) \geqslant \vartheta$ if and only if $\mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$ is nonempty. Obviously, the latter is equivalent to the statement that there exists some $n \in \mathbb{N}$ with $n < \ell$ and $0^n \in \mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$ where $\ell = |Q_1| \cdot \ldots \cdot |Q_k|$.

Let $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ denote the NFA resulting from the union of $\mathcal{D}_1, \ldots, \mathcal{D}_k$. That is, $Q = Q_1 \cup \ldots \cup Q_k$, $Q_0 = \{q_{0,1}, \ldots, q_{0,k}\}$, $F = F_1 \cup \ldots \cup F_k$ and $\delta(q) = \delta_i(q)$ if $q \in Q_i$. That is, besides the nondeterministic choice of the initial state, $\mathcal{A}$ behaves deterministically. The idea is now to treat $\mathcal{A}$ as a substructure of an MDP $\mathcal{M}$ where all states of $\mathcal{M}$ are labeled by $a$. The substructure $\mathcal{A}$ can be entered in $\mathcal{M}$ 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 $\mathcal{M}$. In each state $q \in Q$ an action $\alpha$ is enabled that mimics $\mathcal{A}$'s transition from $q$ with probability $(r-1)/r$ and moves back to the state where $\mathcal{A}$ 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 $\mathcal{M}$ can return to the state where $\mathcal{M}$ can be entered. Other states and transitions of $\mathcal{M}$ are needed for technical purposes.

The idea of this construction is as follows. To maximize the long-run probability for $a \,\mathrm{U}\, b$ in $\mathcal{M}$, a scheduler guesses a natural number $n$ with with $n \leqslant \ell$ and $0^n \in \mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$. It then uses the pumping option to enter $\mathcal{M}$ after having generated a sequence of $\ell - n$ $a$-states. It then attempts to make $n$ steps inside $\mathcal{A}$ using action $\alpha$ and leaves $\mathcal{A}$ using the $\beta$ transition.

Formally, the state space of $\mathcal{M}$ is

$$S \;=\; Q \cup \{a, b, c, \mathit{init}\}$$

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

- In $s \in \{\mathit{init}, a\}$, actions *enter* and *pump* are enabled with the transition probabilities:

  $P(s, \mathit{enter}, q_{0,i}) = \frac{r-1}{k \cdot r}$, $i = 1, \ldots, k$,
  $P(s, \mathit{enter}, \mathit{init}) = \frac{1}{r}$,
  $P(s, \mathit{pump}, a) = \frac{r-1}{r}$, $P(s, \mathit{pump}, \mathit{init}) = \frac{1}{r}$.

- In each state $q \in Q$, action $\alpha$ is enabled with:

  $$P(q, \alpha, \delta(q)) = \tfrac{r-1}{r}, \;\; P(q, \alpha, \mathit{init}) = \tfrac{1}{r}$$

  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:

  $$P(s', \tau, c) = \tfrac{r'-1}{r'}, \quad P(s', \tau, \mathit{init}) = \tfrac{1}{r'}$$

*Scheduler $\mathfrak{S}_n$ and its long-run probability.:* Before defining the values $r$ and $r'$, let us suppose $n$ is an integer with $n < \ell$ and $0^n \in \mathcal{L}(\mathcal{D}_1) \cap \ldots \cap \mathcal{L}(\mathcal{D}_k)$. Regard the FMD-scheduler $\mathfrak{S}_n$ which behaves as follows. Let $m = \ell - n - 1$. From state *init*, $\mathfrak{S}_n$ 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 $\mathfrak{S}_n$ enters $\mathcal{A}$ via *enter*. Let $q_{0,i}$ be the state reached after that sequence. $\mathfrak{S}_n$ then attempts to follow the unique accepting run for $0^n$ in $\mathcal{D}_i$ (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 $\mathfrak{S}_n$ returns to *init* and behaves in the same way.

The long-run probability $\mathbb{LP}^{\mathfrak{S}_n}_{\mathcal{M}}(a \,\mathrm{U}\, b)$ under $\mathfrak{S}_n$ 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 $1 \leqslant k \leqslant \ell$, the scheduler succeeds in generating a path starting with $k$ consecutive $a$-states with probability $\rho^k$. The probability to satisfy $a \,\mathrm{U}\, b$ afterwards is $\rho^{\ell-k}$. In addition $b$ is reached with probability $\rho^\ell$ and here the probability of $a \,\mathrm{U}\, b$ 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 $1 \leqslant k \leqslant \ell$ 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

$$\begin{aligned}
&\sum_{k=1}^{\ell} k(1-\rho)\rho^{k-1} + (\ell+r'+1)\rho^\ell \\
=\;& \frac{1-\rho^\ell}{1-\rho} - \ell\cdot\rho^\ell + (\ell+r'+1)\rho^\ell \\
=\;& r + (r'+1-r)\rho^\ell
\end{aligned}$$

as $\frac{1}{1-\rho} = r$. We let $r' := r-1$ and obtain

$$\mathbb{LP}^{\mathfrak{S}_n}_{\mathcal{M}}(a \,\mathrm{U}\, b) = \frac{\rho^\ell(\ell+1)}{r} = (1-\rho)\rho^\ell(\ell+1).$$

*Definition of* $r$ *(and* $\rho = (r-1)/r$*):* For $j \in \mathbb{N}$, define $\wp_j := (1-\rho)\rho^j(j+1)$. The goal is to choose $r$ in such a way that $\wp_\ell > \wp_j$ 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 it maximum for $j = \ell$. This is achieved by $r = \ell + 1.5$ as can be seen as follows. Let $j \in \mathbb{N}$ and $\bowtie \in \{<,=,>\}$. Then:

$$\begin{aligned}
\rho^j(j+1) \bowtie \rho^{j+1}(j+2) \quad &\text{iff} \quad j+1 \bowtie \tfrac{r-1}{r}(j+2) \\
&\text{iff} \quad \tfrac{j+1}{j+2} \bowtie \tfrac{r-1}{r} \\
&\text{iff} \quad j+2 \bowtie r
\end{aligned}$$

In particular, if $r = \ell+1.5$ then $\ell+2 > r$ and therefore:

$$\wp_\ell > \wp_{\ell+1} > \wp_{\ell+2} > \ldots$$

On the other hand, $(\ell-1)+2 = \ell+1 < r$ and therefore:

$$\wp_\ell > \wp_{\ell-1} > \wp_{\ell-2} > \ldots$$

*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):

$$\vartheta_\ell \;\stackrel{\text{def}}{=}\; \wp_\ell - \frac{\mu}{k}$$

where $\mu$ is the minimal difference between the value $\wp_\ell$ and one the values $\wp_j$ for $j \neq \ell$.

Note that when entering $\mathcal{A}$ after having visited state $a$ $m$-times, then at least one of the DFA $\mathcal{D}_1, \ldots, \mathcal{D}_k$ 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 $\wp_j < \wp_\ell$. Same for combinations of such options depending on which of the states $q_{0,i}$ is entered.

As the values $\wp_j$ are strictly increasing for $j < \ell$ and strictly decreasing for $j > \ell$ we have:

$$\mu \;=\; \min\{\; \wp_\ell - \wp_{\ell-1},\; \wp_\ell - \wp_{\ell+1} \;\}$$

Using the fact that

$$\rho^\ell \;=\; \left(1-\tfrac{1}{r}\right)^\ell \;\geqslant\; 1 - \tfrac{\ell}{r}$$

and using $r = \ell+1.5$ and $(1-\rho) = 1/r$ we obtain:

$$\begin{aligned}
\wp_\ell - \wp_{\ell-1} \;&=\; (1-\rho)(\rho^\ell(\ell+1) - \rho^{\ell-1}\ell) \\
&\geqslant\; \tfrac{1}{r}\left(1-\tfrac{\ell}{r}\right)\left(1+\ell-\ell/\rho\right) \\
&=\; \tfrac{3}{2r^2}\cdot\left(1-\tfrac{\ell}{r-1}\right) \;=\; \tfrac{3}{4r^2(r-1)} \geqslant \tfrac{3}{4r^3}.
\end{aligned}$$

and

$$\begin{aligned}
\wp_\ell - \wp_{\ell+1} \;&=\; (1-\rho)(\rho^\ell(\ell+1) - \rho^{\ell+1}(\ell+2)) \\
&\geqslant\; \tfrac{1}{r}\left(1-\tfrac{\ell}{r}\right)\left(1+\ell(1-\rho)-2\rho\right) \\
&=\; \tfrac{3}{2r^2}\cdot\tfrac{1}{2r} = \tfrac{3}{4r^3}.
\end{aligned}$$

This yields $\mu \geqslant \frac{3}{4r^3}$. In addition, $k \leqslant \log(\ell)$ as we assume that all $\mathcal{D}_i$ have at least two states. To get a safe approximation, we observe that for $\ell \geqslant 1000$, we have $\frac{100}{\ell^4} < \frac{\mu}{k}$.

The crux is now to find a rational threshold $\vartheta$ with $\vartheta_\ell < \vartheta \leqslant \wp_\ell$ that is computable in polynomial time. We cannot use $\wp_\ell$ as its logarithmic length is exponential in the input size (the sum of the sizes of $\mathcal{D}_1, \ldots, \mathcal{D}_k$). Instead, we will compute an approximation $\wp'$ of $\wp_\ell \cdot \ell^4$ up to an absolute error $< 50$ in polynomial time. Then for $\ell \geqslant 1000$, we can choose the threshold $\vartheta$ to be $\frac{\wp'-50}{\ell^4}$ as

$$\wp_\ell \geqslant \frac{\wp'-50}{\ell^4} \geqslant \wp_\ell - \frac{100}{\ell^4} > \wp_\ell - \mu/k = \vartheta_\ell.$$

*Approximation of* $\wp_\ell \cdot \ell^4$*:* We define the real function

$$f(z) := \frac{(1/z+1/2)^{1/z}}{(1/z+3/2)^{1/z+1}}(1/z+1)$$

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 $f(1/\ell) = \wp_\ell$ 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:

$$f(z) = \frac{1}{e} + \frac{z}{2e} - \frac{z^2}{3e} + \frac{z^3}{4e} - \frac{313z^4}{1440e} + \mathcal{O}(z^5)$$

for $z \to 0$. So there are reals $c_0$, $z_0 > 0$ such that

$$\left| f(z) - \left(\frac{1}{e} + \frac{z}{2e} - \frac{z^2}{3e} + \frac{z^3}{4e} - \frac{313z^4}{1440e}\right)\right| \leqslant c_0 z^5$$

for $|z| < z_0$. Using $\wp_\ell = f(1/\ell)$ we obtain

$$\left| \wp_\ell \ell^4 - \frac{1}{e}\left(\ell^4 + \frac{1}{2}\ell^3 - \frac{1}{3}\ell^2 + \frac{1}{4}\ell - \frac{313}{1440}\right)\right| \leqslant c_0 \ell^{-1}$$

for $\ell > 1/z_0$.

Let $\mathsf{L} = \max\{1/z_0, c_0, 1000\}$. To obtain the desired approximation of $\wp_\ell \ell^4$ for $\ell > \mathsf{L}$, 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 $\ell > \mathsf{L}$ which completes the reduction of the intersection problem for unary DFA to the threshold problem for maximal long-run probabilities in MDPs. As $\mathsf{L}$ is defined in terms of the function $\mathsf{f}$, i.e. independent of all variables, and as there are only finitely many instances with $\ell \leqslant \mathsf{L}$, this finishes the proof. $\square$