FOUNDATIONS OF PROBABILITY-RAISING CAUSALITY
IN MARKOV DECISION PROCESSES

CHRISTEL BAIER ⓞ[a], JAKOB PIRIBAUER ⓞ[a], AND ROBIN ZIEMEK ⓞ[a]

Technische Universität Dresden
e-mail address: christel.baier@tu-dresden.de, jakob.piribauer@tu-dresden.de, robin.ziemek@tu-dresden.de

ABSTRACT. This work introduces a novel cause-effect relation in Markov decision processes using the probability-raising principle. Initially, sets of states as causes and effects are considered, which is subsequently extended to regular path properties as effects and then as causes. The paper lays the mathematical foundations and analyzes the algorithmic properties of these cause-effect relations. This includes algorithms for checking cause conditions given an effect and deciding the existence of probability-raising causes. As the definition allows for sub-optimal coverage properties, quality measures for causes inspired by concepts of statistical analysis are studied. These include recall, coverage ratio and f-score. The computational complexity for finding optimal causes with respect to these measures is analyzed.

1. INTRODUCTION

In recent years, scientific and technological advancement in computer science and engineering led to an ever increasing influence of computer systems on our everyday lives. A lot of decisions which were historically done by humans are now in the hands of intelligent systems. At the same time, these systems grow more and more complex, and thus, harder to understand. This poses a huge challenge in the development of reliable and trustworthy systems. Therefore, an important task of computer science today is the broad development of comprehensive and versatile ways to understand modern software and cyber physical systems.

The area of formal verification aims to prove the correctness of a system with respect to a specification. While the formal verification process can provide guarantees on the behavior of a system, such a result alone does not tell much about the inner workings of the system. To give some additional insight, counterexamples, invariants or related certificates as a form of verifiable justification that a system does or does not behave according to a specification have been studied extensively (see e.g., [MP95, CGP99, Nam01]). These kinds of certificates, however, do not allow us to understand the system behavior to a full extend. In epistemic terms, the outcome of model checking applied to a system and a specification provides *knowledge that* a system satisfies a

specification (or not) in terms of an assertion (whether the system satisfies the specification) and a justification (certificate or counterexample) to increase the belief in the result. However, model checking usually does not provide *understanding* on why a system behaves in a certain way. This establishes a desideratum for a more comprehensive understanding of *why* a system satisfies or violates a specification. Explications of the system are needed to assess *how* different components influence its behavior and performance. Causal relations between occurring events during the execution of a system can constitute a strong tool to form such an understanding. Moreover, causality is fundamental for determining moral responsibility [CH04, BvH12] or legal accountability [FHJ⁺11], and ultimately fosters user acceptance through an increased level of transparency [Mil17].

The majority of prior work in this direction relies on causality notions based on Lewis' counter-factual principle [Lew73] which states that the effect would not have occurred if the cause would not have happened. A prominent formalization of the counterfactual principle is given by Halpern and Pearl [HP01] via structural equation models. This inspired formal definitions of causality and related notions of blameworthiness and responsibility in Kripke and game structures as well as reactive systems (see, e.g., [CHK08, BBC⁺12, Cho16, YD16, FH19, YDJ⁺19, BFM21, CFF⁺22]).

A lot of systems are to a certain extend influenced by probabilistic events. Thus, a branch of formal methods is studying probabilistic models such as Markov chains (MCs) which are purely probabilistic or Markov decision processes (MDPs) which combine non-determinism and probabilistic choice. This gives rise to another approach to the concept of causality in a probabilistic setting, since the statement of counterfactual reasoning can be interpreted more gently in a probabilistic setting: Instead of saying "the effect would not have occurred, if the cause had not happened", we can say "the probability of the effect would have been lower if the cause would not have occurred". This interpretation leads to the widely accepted *probability-raising principle* which also has its roots in philosophy [Rei56, Sup70, Eel91, Hit16] and has been refined by Pearl [Pea09] for causal and probabilistic reasoning in intelligent systems. The different notions of probability-raising cause-effect relations discussed in the literature share the following two main principles:

(C1): Causes raise the probabilities for their effects, informally expressed by the requirement "$\Pr(\,\text{effect}\,|\,\text{cause}\,) > \Pr(\,\text{effect}\,)$".

(C2): Causes must happen before their effects.

Despite the huge amount of work on probabilistic causation in other disciplines, research on probability-raising causality in the context of formal methods is comparably rare and has concentrated on the purely probabilistic setting in Markov chains (see, e.g., [KM09, Kle12, ZPF⁺22] and the discussion of related work below). To the best of our knowledge, probabilistic causation for probabilistic operational models with nondeterminism has not been studied before.

In this work, we formalize principles (C1) and (C2) for Markov decision processes. We start in a basic setting by focusing on reachability properties where both effect and cause are sets of states. Later, we naturally extend this framework by considering the effect to be an ω-regular path property while causes can either still be state-based or ω-regular co-safety path properties.

As we like to to have probability-raising inherent in the MDP, we require (C1) under every scheduler. Thus, the cause-effect relation holds for every resolution of the non-deterministic choices. We consider two natural ways to interpret condition (C1): On one hand, the probability-raising property can be required locally for each element of the cause. This results in a strict property which requires that after each execution leading to the cause the probability of effect has been raised. Such causes are called *strict probability-raising (SPR) causes* in our framework. This interpretation is especially suited when the task is to identify system states that have to be avoided for lowering the effect probability. On the other hand, one might want to treat the cause globally as a unit in (C1)

leading to the notion of *global probability-raising (GPR) cause*. This way, the causal relation can also be formulated between properties instead of considering individual elements or executions to be causal. Considering the cause as a whole is also better suited when further constraints are imposed on the candidates for cause sets. E.g. if the set of non-terminal states of the given MDP is partitioned into sets of states S_i under control of different agents i, $1 \leqslant i \leqslant k$ and the task is to identify which agent's decisions might cause the effect, only the subsets of $S_1, \ldots, S_k$ are candidates for causes. Furthermore, global causes are more appropriate when causes are used for monitoring purposes under partial observability constraints as then the cause candidates are sets of indistinguishable states.

Even with the distinction between strict and global probability-raising causality, different causes might still vary substantially regarding how well they predict the effect. Within Markov decision processes this intuitively coincides with how well the executions exhibiting the cause cover executions showing the effect. However, solely focusing on broader coverage might also lead to more trivial causal relations. In order to take this trade off into account, we take inspiration from measures for binary classifiers used in statistical analysis (see, e.g., [Pow11]) to introduce quality measures for causes. These allow us to compare different causes and to look for optimal causes: The *recall* captures the probability that the effect is indeed preceded by the cause. The *coverage-ratio* quantifies the fraction of the probability that cause and effect are observed and the probability that the effect but not the cause is observed. The *f-score*, a widely used quality measure for binary classifiers, is the harmonic mean of recall and precision, where the precision is the probability that the cause is followed by the effect. Finally, we address the computation of arbitrary quality measures as long as they can be represented as algebraic functions.

Contributions. In this work we build mathematical and algorithmic foundations for probabilistic causation in Markov decision processes based on the principles (C1) and (C2). In the setting where the effect is represented as a set of terminal states, we introduce strict and global probability-raising cause sets in MDPs (Section 3). Algorithms are provided to check whether given cause and effect sets satisfy (one of) the probability-raising conditions (Section 4.1 and 4.2) and to check the existence of a cause set for a given effect set (Section 4.1). In order to evaluate the coverage properties of a cause, we subsequently introduce the above-mentioned quality measures (Section 5.1). We give algorithms for computing these values for given cause-effect sets (Section 5.2) and characterize the computational complexity of finding optimal cause sets with respect to the different measures (Section 5.3). We then extend the setting to ω-regular effects (Section 6.1), and evaluate how established properties transfer to this setting. We observe that in this extension SPR causes can be viewed as a collection of GPR causes. Finally we discuss the case where causes are also path properties, namely, ω-regular co-safety properties and investigate how this more general perspective affects cause-effect relations (Section 6.2). Here, the class of potential cause candidates is greatly increased. Table 1 summarizes our complexity results.

Related work. Previous work in the direction of probabilistic causation in stochastic operational models has mainly concentrated on Markov chains. Kleinberg [KM09, Kle12] introduced *prima facie causes* in finite Markov chains where both causes and effects are formalized as PCTL state formulae, and thus they can be seen as sets of states as in our approach. The correspondence of Kleinberg's PCTL constraints for prima facie causes and the strict probability-raising condition formalized using conditional probabilities has been worked out in the survey article [BDF$^+$21]. Our notion of strict probability-raising causes interpreted in Markov chains corresponds to Kleinberg's prima facie causes with the exception of a minimality condition forbidding redundant elements in our definition.

Table 1: Complexity results for MDPs and Markov chains (MC) with fixed effects. The ω-regular effects are given as deterministic Rabin automata, ω-regular co-safety properties as causes are given as deterministic finite automata accepting good prefixes.

sets of states as causes and effects

| | for fixed set Cause | | find optimal cause | |
	check PR condition	compute quality values (recall, covratio, f-score)	covratio-optimal = recall-optimal	f-score-optimal
SPR	$\in$ P	poly-time	poly-time	poly-space poly-time for MC threshold problem $\in$ NP$\cap$coNP
GPR	$\in$ coNP and $\in$ P for MC	poly-time	poly-space threshold problems $\in \Sigma_2^P$ and NP-hard and NP-complete for MC	

sets of states as causes and ω-regular effects

| | for fixed set Cause | | find optimal cause | |
	check PR condition	compute quality values (recall, covratio, f-score)	covratio-optimal = recall-optimal	f-score-optimal
SPR	$\in$ coNP and $\in$ P for MC	poly-time	PF^{NP} (as def. in [Sel94]) threshold problems $\in$ coNP	poly-space threshold problem $\in \Sigma_2^P$
GPR	$\in$ coNP and $\in$ P for MC	poly-time	poly-space threshold problems $\in \Sigma_2^P$ NP-hard	

ω-regular co-safety properties as causes and ω-regular effects

| | for fixed cause property rCause | | find optimal cause | |
	check PR condition	compute quality values (recall, covratio, f-score)	covratio-optimal = recall-optimal	f-score-optimal
SPR	difficulty illustrated in Example 6.14	poly-time	optimal cause known, computation unclear	open
GPR	$\in$ coNP and $\in$ P for MC	poly-time	in general, no optimal causes	open

Ábrahám et al [ÁB18] introduces a hyperlogic for Markov chains and gives a formalization of probabilistic causation in Markov chains as a hyperproperty, which is consistent with Kleinberg's prima facie causes, and with strict probability-raising causes up to minimality. Cause-effect relations in Markov chains where effects are ω-regular properties and the causes are sets of paths have been introduced in [ZPF$^+$22]. These relations rely on a strict probability-raising condition, but use a probability threshold p instead of directly requiring probability-raising. Therefore, [ZPF$^+$22] permits a non-strict inequality in the PR condition with the consequence that causes always exist, which is not the case for our notions. However, a minimal good prefix of a co-safety strict probability-raising cause in a Markov chains corresponds to a probability-raising path in [ZPF$^+$22].

The survey article [BDF$^+$21] introduces notions of global probability-raising causes for Markov chains, where causes and effects can be path properties. [BDF$^+$21]'s notion of *reachability causes* in Markov chains directly corresponds to our notion GPR causes, the only difference being that [BDF$^+$21] deals with a relaxed minimality condition and requires that the cause set is reachable without visiting an effect state before. The latter is inherent in our approach as we suppose that all states are reachable and the effect states are terminal. On the other hand if we restrict [BDF$^+$21]'s

notion of global PR-cause to ω-regular effects and co-safety causes, this corresponds to our notion of a co-safety GPR cause with the exception of the minimality condition. The same can be said about the correspondence of local PR-causes from [BDF$^+$21] and co-safety SPR causes.

To the best of our knowledge, probabilistic causation in MDPs has not been studied before. The only work in this direction we are aware of is the recent paper by Dimitrova et al [DFT20] on a hyperlogic, called PHL, for MDPs. While the paper focuses on the foundation of PHL, it contains an example illustrating how action causality can be formalized as a PHL formula. Roughly, the presented formula expresses that taking a specific action α increases the probability for reaching effect states. Thus, it also relies on the probability-raising principle, but compares the "effect probabilities" under different schedulers (which either schedule α or not) rather than comparing probabilities under the same scheduler as in our PR condition. However, to some extent our notions of PR causes can reason about action causality as well.

There has also been work on causality-based explanations of counterexamples in probabilistic models [KLL11, Lei15]. The underlying causality notion of this work, however, relies on the non-probabilistic counterfactual principle rather than the probability-raising condition. The same applies to the notions of forward and backward responsibility in stochastic games in extensive form introduced in the recent work [BFM21].

2. PRELIMINARIES

Throughout the paper, we will assume familiarity with basic concepts of Markov decision processes. Here, we present a brief summary of the notations used in the paper. For more details, we refer to [Put94, BK08, Kal20].

2.1. **Markov decision processes.** A *Markov decision process (MDP)* is a tuple $\mathcal{M} = (S, Act, P, \text{init})$ where S is a finite set of states, Act a finite set of actions, $\text{init} \in S$ a unique initial state and $P : S \times Act \times S \to [0,1]$ the transition probability function such that $\sum_{t \in S} P(s, \alpha, t) \in \{0, 1\}$ for all states $s \in S$ and actions $\alpha \in Act$.

For $\alpha \in Act$ and $T \subseteq S$, $P(s, \alpha, T)$ is a shortform notation for $\sum_{t \in t} P(s, \alpha, t)$. An action α is *enabled* in state $s \in S$ if $\sum_{t \in S} P(s, \alpha, t) = 1$. We define $Act(s) = \{\alpha \in Act \mid \alpha \text{ is enabled in } s\}$. A state t is *terminal* if $Act(t) = \emptyset$. A Markov chain (MC) is a special case of MDP where Act is a singleton (we then write $P(s, t)$ rather than $P(s, \alpha, t)$). A *path* in an MDP $\mathcal{M}$ is a (finite or infinite) alternating sequence $\pi = s_0 \alpha_0 s_1 \alpha_1 s_2 \cdots \in (S \times Act)^* \times S \cup (S \times Act)^\omega$ such that $P(s_i, \alpha_i, s_{i+1}) > 0$ for all indices i. A path is called maximal if it is infinite or finite and ends in a terminal state. If π is a finite path in $\mathcal{M}$ then $last(\pi)$ denotes the last state of π. That is, if $\pi = s_0 \alpha_0 \ldots \alpha_{n-1} s_n$ then $last(\pi) = s_n$.

A *(randomized) scheduler* $\mathfrak{S}$ is a function that maps each finite non-maximal path $s_0 \alpha_0 \ldots s_n$ to a distribution over $Act(s_n)$. $\mathfrak{S}$ is called deterministic if $\mathfrak{S}(\pi)$ is a Dirac distribution for all finite non-maximal paths π. If the chosen action only depends on the last state of the path, $\mathfrak{S}$ is called *memoryless*. We write MR for the class of memoryless (randomized) and MD for the class of memoryless deterministic schedulers. *Finite-memory* schedulers are those that are representable by a finite-state automaton. A path π is said to be a $\mathfrak{S}$-path if $\mathfrak{S}(s_0 \alpha_0 \ldots \alpha_{i-1} s_i)(\alpha_i) > 0$ for each $i \in \{0, \ldots, n-1\}$. Given a path $\pi = s_0 \alpha_0 \ldots \alpha_{n-1} s_n$, the *residual scheduler res*$(\mathfrak{S}, \pi)$ of $\mathfrak{S}$ after π is defined by $res(\mathfrak{S}, \pi)(\zeta) = \mathfrak{S}(\pi \circ \zeta)$ for all finite paths ζ starting in s_n. Here, $\pi \circ \zeta$ denotes the concatenation of the paths π and ζ. Intuitively speaking, $res(\mathfrak{S}, \pi)$ behaves like $\mathfrak{S}$ after π has already been seen.

A scheduler $\mathfrak{S}$ of $\mathcal{M}$ induces a (possibly infinite) Markov chain. We write $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}$ for the standard probability measure on measurable sets of maximal paths in the Markov chain induced by $\mathfrak{S}$ with initial state s. If φ is a measurable set of maximal paths, then $\mathrm{Pr}^{\max}_{\mathcal{M},s}(\varphi)$ and $\mathrm{Pr}^{\min}_{\mathcal{M},s}(\varphi)$ denote the supremum resp. infimum of the probabilities for φ under all schedulers. We use the abbreviation $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}} = \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},\mathrm{init}}$ and notations $\mathrm{Pr}^{\max}_{\mathcal{M}}$ and $\mathrm{Pr}^{\min}_{\mathcal{M}}$ for extremal probabilities. Analogous notations will be used for expectations. So, if f is a random variable, then, e.g., $\mathrm{E}^{\mathfrak{S}}_{\mathcal{M}}(f)$ denotes the expectation of f under $\mathfrak{S}$ and $\mathrm{E}^{\max}_{\mathcal{M}}(f)$ its supremum over all schedulers. We also use conditional probabilities in MDPs cf. [BKKM14, Mär20]. For two measurable sets of maximal paths φ and ψ we have

$$\mathrm{Pr}^{\max}_{\mathcal{M},s}(\varphi \mid \psi) = \max_{\mathfrak{S}} \ \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi \mid \psi) = \max_{\mathfrak{S}} \ \frac{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}(\varphi \wedge \psi)}{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M},s}(\psi)},$$

where $\mathfrak{S}$ ranges over all scheduler for which $\mathrm{Pr}^{\mathfrak{S}}(\psi) > 0$. We define $\mathrm{Pr}^{\min}_{\mathcal{M},s}(\varphi \mid \psi)$ likewise. If both φ and ψ are reachability properties then maximal conditional probabilities are computable in polynomial time [BKKM14]. The proposed algorithm for maximal conditional probabilities $\mathrm{Pr}^{\max}_{\mathcal{M},\mathrm{init}}(\Diamond G \mid \Diamond F)$ relies on a model transformation generating a new MDP $\mathcal{N}$ that distinguishes the modes "before G and F" (where $\mathcal{N}$ essentially behaves as $\mathcal{M}$ with additional reset transitions from end components to the initial state), "before G, after F" (where $\mathcal{N}$ maximizes the probability to reach G), "before F, after G" (where $\mathcal{N}$ maximizes the probability to reach F). Essentially the reset transitions serve to "discard" paths that never reach G and F. For further details we refer to [BKKM14, MBKK17, Mär20].

If $s \in S$ and $\alpha \in Act(s)$, then (s, α) is said to be a state-action pair of $\mathcal{M}$. Given a scheduler $\mathfrak{S}$ for $\mathcal{M}$, the expected frequencies (i.e., expected number of occurrences in maximal paths) of state action-pairs (s, α), states $s \in S$ and state-sets $T \subseteq S$ under $\mathfrak{S}$ are defined by:

$$freq_{\mathfrak{S}}(s, \alpha) \ \stackrel{\mathrm{def}}{=} \ \mathrm{E}^{\mathfrak{S}}_{\mathcal{M}}(\text{number of visits to } s \text{ in which } \alpha \text{ is taken})$$

$$freq_{\mathfrak{S}}(s) \ \stackrel{\mathrm{def}}{=} \sum_{\alpha \in Act(s)} freq_{\mathfrak{S}}(s, \alpha), \qquad freq_{\mathfrak{S}}(T) \stackrel{\mathrm{def}}{=} \sum_{s \in T} freq_{\mathfrak{S}}(s).$$

We use LTL-like temporal modalities such as $\Diamond$ (eventually) and U (until) to denote path properties. For $X, T \subseteq S$ the formula $X \mathrm{U} T$ is satisfied by paths $\pi = s_0 s_1 \ldots$ such that there exists $j \geqslant 0$ such that for all $i < j : s_i \in X$ and $s_j \in T$ and $\Diamond T = S \mathrm{U} T$. It is well-known that $\mathrm{Pr}^{\min}_{\mathcal{M}}(X \mathrm{U} T)$ and $\mathrm{Pr}^{\max}_{\mathcal{M}}(X \mathrm{U} T)$ and corresponding optimal MD-schedulers are computable in polynomial time.

An *end component* (EC) of an MDP $\mathcal{M}$ is a strongly connected sub-MDP containing at least one state-action pair. ECs will be often identified with the set of their state-action pairs. An EC $\mathcal{E}$ is called maximal (abbreviated MEC) if there is no proper superset $\mathcal{E}'$ of (the set of state-action pairs of) $\mathcal{E}$ which is an EC.

2.2. **MR-scheduler in MDPs without ECs.** The following preliminary lemma is folklore (see, e.g., [Kal20, Theorem 9.16]) and used in the paper in the following form.

Lemma 2.1 (From general schedulers to MR-schedulers in MDPs without ECs)**.** *Consider an MDP* $\mathcal{M} = (S, Act, P, \mathrm{init})$ *without end components. Then, for each scheduler $\mathfrak{S}$ for $\mathcal{M}$, there exists an MR-scheduler $\mathfrak{T}$ such that:*

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond t) \ = \ \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}}(\Diamond t) \ \ \textit{for each terminal state } t.$$

As a consequence we can build linear combinations of scheduler in such MDPs.

Lemma 2.2 (Convex combination of MR-schedulers). *Let $\mathcal{M}$ be an MDP without end components and let $\mathfrak{S}$ and $\mathfrak{T}$ be schedulers for $\mathcal{M}$ and λ a real number in the open interval $]0,1[$. Then, there exists an MR-scheduler $\mathfrak{U}$ such that:*

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\Diamond t) \;=\; \lambda \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond t) \;+\; (1-\lambda) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond t)$$

for each terminal state t.

Proof. Thanks to Lemma 2.1 we may suppose that $\mathfrak{S}$ and $\mathfrak{T}$ are MR-schedulers. Let

$$f_* \;=\; \lambda \cdot freq_{\mathfrak{S}}(*) + (1-\lambda) \cdot freq_{\mathfrak{T}}(*)$$

where $*$ stands for a state or a state-action pair in $\mathcal{M}$. Let $\mathfrak{U}$ be an MR-scheduler defined by $\mathfrak{U}(s)(\alpha) = \frac{f_{s,\alpha}}{f_s}$ for each non-terminal state s where $f_s > 0$ and each action $\alpha \in Act(s)$. If $f_s = 0$ then $\mathfrak{U}$ selects an arbitrary distribution over $Act(s)$.

Using Lemma 2.1 we then obtain $f_* = freq_{\mathfrak{U}}(*)$ where $*$ ranges over all states and state-action pairs in $\mathcal{M}$. But this yields:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\Diamond t) \;=\; f_t \;=\; \lambda \cdot freq_{\mathfrak{S}}(t) + (1-\lambda) \cdot freq_{\mathfrak{T}}(t) \;=\; \lambda \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond t) + (1-\lambda) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond t)$$

for each terminal state t. $\qquad\square$

Let $\mathcal{M}, \mathfrak{S}, \mathfrak{T}, \lambda$ be as in Lemma 2.2. Then, the notation $\lambda\mathfrak{S} \oplus (1-\lambda)\mathfrak{T}$ will be used to denote any MR-scheduler $\mathfrak{U}$ as in Lemma 2.2.

2.3. MEC-quotient.
We recall the definition of the MEC-quotient, which is a standard concept for the analysis of MDPs [dA97]. Intuitively, the MEC-quotient of an MDP collapses all maximal end components, ignoring the actions in the end component while keeping outgoing transitions. More concretely, we use a modified version with an additional trap state as in [BBD+18] that serves to mimic behaviors inside an end component of the original MDP.

Definition 2.3 (MEC-quotient of an MDP). Let $\mathcal{M} = (S, Act, P, \mathrm{init})$ be an MDP with end components. Let $\mathcal{E}_1, \ldots, \mathcal{E}_k$ be the maximal end components (MECs) of $\mathcal{M}$. We may suppose without loss of generality that enabled actions of states are pairwise disjoint, i.e., whenever s_1, s_2 are states in $\mathcal{M}$ with $s_1 \neq s_2$ then $Act_{\mathcal{M}}(s_1) \cap Act_{\mathcal{M}}(s_2) = \varnothing$. This permits to consider $\mathcal{E}_i$ as a subset of Act. Let U_i denote the set of states that belong to $\mathcal{E}_i$ and let $U = U_1 \cup \ldots \cup U_k$. The *MEC-quotient of* $\mathcal{M}$ is the MDP $\mathcal{N} = (S', Act', P', \mathrm{init}')$ and the function $\iota \colon S \to S'$ are defined as follows.

- The state space S' is $S \setminus U \cup \{s_{\mathcal{E}_1}, \ldots, s_{\mathcal{E}_k}, \bot\}$ where $s_{\mathcal{E}_1}, \ldots, s_{\mathcal{E}_k}, \bot$ are pairwise distinct fresh states.
- The function ι is given by $\iota(s) = s$ if $s \in S \setminus U$ and $\iota(u) = s_{\mathcal{E}_i}$ if $u \in U_i$.
- The initial state of $\mathcal{N}$ is $\mathrm{init}' = \iota(\mathrm{init})$.
- The action set Act' is $Act \cup \{\tau\}$ where τ is a fresh action symbol.
- The set of actions enabled in state $s \in S'$ of $\mathcal{N}$ and the transition probabilities are as follows:
  - If s is a state of $\mathcal{M}$ that does not belong to an MEC of $\mathcal{M}$ (i.e., $s \in S \cap S'$) then $Act_{\mathcal{N}}(s) = Act_{\mathcal{M}}(s)$ and $P'(s, \alpha, s') = P(s, \alpha, \iota^{-1}(s'))$ for all $s' \in S'$ and $\alpha \in Act_{\mathcal{M}}(s)$.
  - If $s = s_{\mathcal{E}_i}$ is a state representing MEC $\mathcal{E}_i$ of $\mathcal{M}$ then (as we view $\mathcal{E}_i$ as a set of actions):

  $$Act_{\mathcal{N}}(s_{\mathcal{E}_i}) \;=\; \bigcup_{u \in U_i} (Act_{\mathcal{M}}(u) \setminus \mathcal{E}_i) \cup \{\tau\}$$

  The τ-action stands for the deterministic transition to the fresh state $\bot$, i.e.: $P'(s_{\mathcal{E}_i}, \tau, \bot) = 1$. For $u \in U_i$ and $\alpha \in Act_{\mathcal{M}}(u) \setminus \mathcal{E}_i$ we set $P'(s_{\mathcal{E}_i}, \alpha, s') = P(u, \alpha, \iota^{-1}(s'))$ for all $s' \in S'$.
  - The state $\bot$ is terminal, i.e., $Act_{\mathcal{N}}(\bot) = \varnothing$. $\qquad\triangleleft$

Thus, each terminal state of $\mathcal{M}$ is terminal in its MEC-quotient $\mathcal{N}$ too. Vice versa, every terminal state of $\mathcal{N}$ is either a terminal state of $\mathcal{M}$ or $\bot$. Moreover, $\mathcal{N}$ has no end components, which implies that under every scheduler $\mathfrak{T}$ for $\mathcal{N}$, a terminal state will be reached with probability 1. In Section 4.2, we use the notation $\mathsf{noeff}_{\mathsf{tn}}$ rather than $\bot$.

The original MDP and its MEC-quotient have been found to be connected by the following lemma (see also [dA97, dA99]). For the sake of completeness we present the proof for our version of the MEC-quotient.

Lemma 2.4 (Correspondence of an MDP and its MEC-quotient). *Let $\mathcal{M}$ be an MDP and $\mathcal{N}$ its MEC-quotient. Then, for each scheduler $\mathfrak{S}$ for $\mathcal{M}$ there is a scheduler $\mathfrak{T}$ for $\mathcal{N}$ such that*

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{t}) \;=\; \mathrm{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond\mathsf{t}) \;\textit{for each terminal state}\ \mathsf{t}\ \textit{of}\ \mathcal{M} \tag{2.1}$$

and vice versa. Moreover, if (2.1) holds then $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond\bot)$ equals the probability for $\mathfrak{S}$ to generate an infinite path in $\mathcal{M}$ that eventually enters and stays forever in an end component.

Proof. Given a scheduler $\mathfrak{T}$ for $\mathcal{N}$, we pick an MD-scheduler $\mathfrak{U}$ such that $\mathfrak{U}(\mathsf{u}) \in \mathcal{E}_i$ for each $\mathsf{u} \in U_i$. Then, the corresponding scheduler $\mathfrak{S}$ for $\mathcal{M}$ behaves as $\mathfrak{T}$ as long as $\mathfrak{T}$ does not choose the τ-transition to $\bot$. As soon as $\mathfrak{T}$ schedules τ then $\mathfrak{S}$ behaves as $\mathfrak{U}$ from this moment on.

Vice versa, given a scheduler $\mathfrak{S}$ for $\mathcal{M}$ then a corresponding scheduler $\mathfrak{T}$ for $\mathcal{N}$ mimics $\mathfrak{S}$ as long as $\mathfrak{S}$ has not visited a state belong to an end component $\mathcal{E}_i$ of $\mathcal{M}$. Scheduler $\mathfrak{T}$ ignores $\mathfrak{S}$'s transitions inside an MEC $\mathcal{E}_i$ and takes $\beta \in \bigcup_{\mathsf{u} \in U_i}(Act_{\mathcal{M}}(\mathsf{u}) \setminus \mathcal{E}_i)$ with the same probability as $\mathfrak{S}$ leaves $\mathcal{E}_i$. With the remaining probability mass, $\mathfrak{S}$ stays forever inside $\mathcal{E}_i$, which is mimicked by $\mathfrak{T}$ by taking the τ-transition to $\bot$.

For the formal definition of $\mathfrak{T}$, we use the following notation. For simplicity, let us assume that $\mathsf{init} \notin U_1 \cup \ldots \cup U_k$. This yields $\mathsf{init} = \mathsf{init}'$. Given a finite path $\pi = s_0\,\alpha_0\,s_1\,\alpha_1\ldots\alpha_{m-1}\,s_m$ in $\mathcal{M}$ with $s_0 = \mathsf{init}$, let $\pi_{\mathcal{N}}$ the path in $\mathcal{N}$ resulting from by replacing each maximal path fragment $s_h\,\alpha_h\ldots\alpha_{j-1}\,s_j$ consisting of actions inside an $\mathcal{E}_i$ with state $s_{\mathcal{E}_i}$. (Here, maximality means if $h > 0$ then $\alpha_{h-1} \notin \mathcal{E}_i$ and if $j < m$ then $\alpha_{j+1} \notin \mathcal{E}_i$.) Furthermore, let $p_\pi^{\mathfrak{S}}$ denote the probability for $\mathfrak{S}$ to generate the path π when starting in the first state of π.

Let ρ be a finite path in $\mathcal{N}$ with first state init (recall that we suppose that $\mathcal{M}$'s initial state does not belong to an MEC, which yields $\mathsf{init} = \mathsf{init}'$) and $last(\rho) \neq \bot$. Then, Π_ρ denotes the set of finite paths $\pi = s_0\,\alpha_0\,s_1\,\alpha_1\ldots\alpha_{m-1}\,s_m$ in $\mathcal{M}$ such that (i) $\pi_{\mathcal{N}} = \rho$ and (ii) if $s_m \in U_i$ then $\alpha_{m-1} \notin \mathcal{E}_i$. The formal definition of scheduler $\mathfrak{T}$ is as follows. Let ρ be a finite path in $\mathcal{N}$ where the last state s of ρ is non-terminal. If s is a state of $\mathcal{M}$ and does not belong to an MEC of $\mathcal{M}$ and $\beta \in Act_{\mathcal{M}}(s)$ then:

$$\mathfrak{T}(\rho)(\beta) \;=\; \sum_{\pi \in \Pi_\rho} p_\pi^{\mathfrak{S}} \cdot \mathfrak{S}(\pi)(\beta)$$

If $s = s_{\mathcal{E}_i}$ and $\beta \in Act_{\mathcal{N}}\big(s_{\mathcal{E}_i}\big) \setminus \{\tau\}$ then

$$\mathfrak{T}(\rho)(\beta) \;=\; \sum_{\pi \in \Pi_\rho} p_\pi^{\mathfrak{S}} \cdot \mathrm{Pr}_{\mathcal{M},last(\pi)}^{res(\mathfrak{S},\pi)}\big(\text{``leave } \mathcal{E}_i \text{ via action } \beta\text{''}\big)$$

where "leave $\mathcal{E}_i$ via action β" means the existence of a prefix whose action sequence consists of actions inside $\mathcal{E}_i$ followed by action β. The last state of this prefix, however, could be a state of U_i. (Note $\beta \in Act_{\mathcal{N}}(s_{\mathcal{E}_i})$ means that β could have reached a state outside U_i, but there might be states inside U_i that are accessible via β.) Similarly,

$$\mathfrak{T}(\rho)(\tau) \;=\; \sum_{\pi \in \Pi_\rho} p_\pi^{\mathfrak{S}} \cdot \mathrm{Pr}_{\mathcal{M},last(\pi)}^{res(\mathfrak{S},\pi)}\big(\text{``stay forever in } \mathcal{E}_i\text{''}\big)$$

where "stay forever in $\mathcal{E}_i$" means that only actions inside $\mathcal{E}_i$ are performed. By induction on the length of ρ we obtain:

$$p_\rho^{\mathfrak{T}} \;=\; \sum_{\pi \in \Pi_\rho} p_\pi^{\mathfrak{S}}$$

But this yields $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond t) = \mathrm{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond t)$ for each terminal state t of $\mathcal{M}$. Moreover the probability under $\mathfrak{S}$ to eventually enter and stay forever in $\mathcal{E}_i$ equals the probability for $\mathfrak{T}$ to reach the terminal state $\perp$ via a path of the form $\rho\tau\perp$ where $last(\rho) = s_{\mathcal{E}_i}$. $\qquad\square$

2.4. **Automata and ω-regular languages.** In order have a representation of an ω-regular language, we use *deterministic Rabin automata* (DRA). A DRA is a tuple $\mathcal{A} = (Q, \Sigma, q_0, \delta, \mathrm{Acc})$ where Q is a finite set of states, Σ an alphabet, q_0 the initial state, $\delta : Q \times \Sigma \to Q$ the transition function and $\mathrm{Acc} \subseteq 2^Q \times 2^Q$ the acceptance set. The *run* of $\mathcal{A}$ on a word $w = w_0 w_1 \cdots \in \Sigma^\omega$ is the sequence $\tau = q_0 q_1 \ldots$ of states such that $\delta(q_i, w_i) = q_{i+1}$ for all i. It is *accepting* if there exists a pair $(L, K) \in \mathrm{Acc}$ such that L is only visited finitely often and K is visited infinitely often by τ. The language $\mathcal{L}(\mathcal{A})$ is the set of all words $w \in \Sigma^\omega$ on which the run of $\mathcal{A}$ is accepting.

A *good prefix* π for an ω-regular language $\mathcal{L}$ is a finite word such that all infinite extensions of π belong to $\mathcal{L}$. An ω-regular language $\mathcal{L}$ is called a *co-safety language* if all words in $\mathcal{L}$ have a prefix that is a good prefix for $\mathcal{L}$. A co-safety language $\mathcal{L}$ is uniquely determined by the regular set of minimal good prefixes of words in $\mathcal{L}$.

The regular language of minimal good prefixes of a co-safety $\mathcal{L}$ which uniquely determines $\mathcal{L}$ can be represented by a *deterministic finite automaton* (DFA). A DFA is a tuple $\mathcal{A} = (Q, \Sigma, q_0, \delta, \mathrm{Acc})$ where Q is a finite set of states, Σ an alphabet, q_0 the initial state, $\delta : Q \times \Sigma \to Q$ the transition function and $\mathrm{Acc} \subseteq Q$ the acceptance set. The *run* of $\mathcal{A}$ on a finite word $w = w_0 w_1 \ldots w_n$ is the sequence $\tau = q_0 q_1 \ldots q_n$ of states such that $\delta(q_i, w_i) = q_{i+1}$ for all i. It is *accepting* if $q_n \in \mathrm{Acc}$. The language $\mathcal{L}(\mathcal{A})$ is the set of all words $w \in \Sigma^*$ on which the run of $\mathcal{A}$ is accepting.

Given an MDP $\mathcal{M} = (S, Act, P, \mathrm{init})$ and a DFA $\mathcal{A} = (Q, \Sigma, q_0, \delta, Acc)$ with $\Sigma \subseteq S^*$ we define the *product MDP* $\mathcal{M} \otimes \mathcal{A} = (S \times Q, Act, P', \mathrm{init}')$ with $P'(<s, q>, \alpha, <t, r>) = P(s, \alpha, t)$ if $r = \delta(q, s)$ and 0 otherwise, and $\mathrm{init}' = \delta(q_0, \mathrm{init})$. The same construction works for the product of an MDP with a DRA. The difference comes from the acceptance condition encoded in the second components of states of the product MDP.

3. STRICT AND GLOBAL PROBABILITY-RAISING CAUSES

Our contribution starts by providing novel formal definitions for cause-effect relations in MDPs which rely on the probability-raising (PR) principle $P(E \mid C) > P(C)$ (C1) which states that the probability of the effect is higher after the cause. Additionally, we include temporal priority of causes (C2), stating that causes must happen before the effect. Here, we focus on the case where both causes and effects are state properties, i.e., sets of states.

In the sequel, let $\mathcal{M} = (S, Act, P, \mathrm{init})$ be an MDP and $\mathrm{Eff} \subseteq S \setminus \{\mathrm{init}\}$ a nonempty set of terminal states. As the effect set is fixed, the assumption that all effect states are terminal contributes to the temporal priority (C2). We may also assume that every state $s \in S$ is reachable from init.

We consider two variants of the probability-raising condition: the global setting treats the set Cause as a unit, while the strict view requires (C1) for all states in Cause individually.

Definition 3.1 (Global and strict probability-raising cause (GPR/SPR cause)). Let $\mathcal{M}$ and Eff be as above and Cause a nonempty subset of $S \setminus Eff$ such that for each $c \in$ Cause we have $\mathrm{Pr}_{\mathcal{M}}^{\max}((\neg \mathsf{Cause})\, \mathsf{U}\, c) > 0$. Then, Cause is said to be a *GPR cause* for Eff iff the following condition (G) holds:

(G) : For each scheduler $\mathfrak{S}$ where $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) > 0$:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond \mathsf{Eff} \mid \Diamond \mathsf{Cause}\,) \; > \; \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}). \tag{GPR}$$

Cause is called an *SPR cause* for Eff iff the following condition (S) holds:

(S) : For each state $c \in$ Cause and each scheduler $\mathfrak{S}$ where $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \mathsf{Cause})\, \mathsf{U}\, c) > 0$:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond \mathsf{Eff} \mid (\neg \mathsf{Cause})\, \mathsf{U}\, c\,) \; > \; \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}). \tag{SPR}$$

Note that we only consider sets Cause as PR cause when each state in $c \in$ Cause is accessible from init without traversing other states in Cause. This can be seen as a minimality condition ensuring that a cause does not contain redundant elements. However, we could omit this requirement without affecting the covered effects (events where an effect state is reached after visiting a cause state) or uncovered effects (events where an effect state is reached without visiting a cause state before). More concretely, whenever a set $C \subseteq S \setminus Eff$ satisfies conditions (G) or (S) then the set of states $c \in C$ where $\mathcal{M}$ has a path from init satisfying $(\neg C)\, \mathsf{U}\, c$ is a GPR resp. an SPR cause. On the other hand the set Cause is disjoint of the effect Eff to ensure temporal priority (C2).

3.1. **Examples and simple properties of probability-raising causes.** We first observe that SPR and GPR causes cannot contain the initial state init, since otherwise an equality instead of an inequality would hold in (GPR) and (SPR). Furthermore as a direct consequence of the definitions and using the equivalence of the LTL formulas $\Diamond \mathsf{Cause}$ and $(\neg \mathsf{Cause})\, \mathsf{U}\, \mathsf{Cause}$ we obtain:

Lemma 3.2 (Singleton PR causes). *If* Cause *is a singleton then* Cause *is a SPR cause for* Eff *if and only if* Cause *is a GPR cause for* Eff.

The direction from SPR cause to GPR cause even holds in general as the event $\Diamond \mathsf{Cause}$ can be expressed as a disjoint union of all events $(\neg \mathsf{Cause})\, \mathsf{U}\, c$ where $c \in$ Cause. Therefore, the probability for covered effects $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond \mathsf{Eff} \mid \Diamond \mathsf{Cause}\,)$ is a weighted average of the probabilities $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond \mathsf{Eff} \mid (\neg \mathsf{Cause})\, \mathsf{U}\, c\,)$ for $c \in$ Cause, which yields:

Lemma 3.3. *Every SPR cause for* Eff *is a GPR cause for* Eff.

Proof. Assume that Cause is a SPR cause for Eff in $\mathcal{M}$ and let $\mathfrak{S}$ be a scheduler that reaches Cause with positive probability. Further, let

$$C_{\mathfrak{S}} \stackrel{\text{def}}{=} \{c \in \mathsf{Cause} \mid \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \mathsf{Cause})\, \mathsf{U}\, c) > 0\} \quad \text{and} \quad \mathfrak{m} \stackrel{\text{def}}{=} \min_{c \in C_{\mathfrak{S}}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond \mathsf{Eff} \mid (\neg \mathsf{Cause})\, \mathsf{U}\, c\,).$$

As Cause is a SPR cause, $\mathfrak{m} > \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff})$. The set of $\mathfrak{S}$-paths satisfying $\Diamond \mathsf{Cause}$ is the disjoint union of the sets of $\mathfrak{S}$-paths satisfying $(\neg \mathsf{Cause})\, \mathsf{U}\, c$ with $c \in C_{\mathfrak{S}}$. Hence,

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff} \mid \Diamond \mathsf{Cause}) = \frac{\sum_{c \in C_{\mathfrak{S}}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff} \mid (\neg \mathsf{Cause})\, \mathsf{U}\, c) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \mathsf{Cause})\, \mathsf{U}\, c)}{\sum_{c \in C_{\mathfrak{S}}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \mathsf{Cause})\, \mathsf{U}\, c)} \geqslant \mathfrak{m}.$$

As $\mathfrak{m} > \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff})$, the GPR condition (GPR) is satisfied under $\mathfrak{S}$. $\qquad \square$
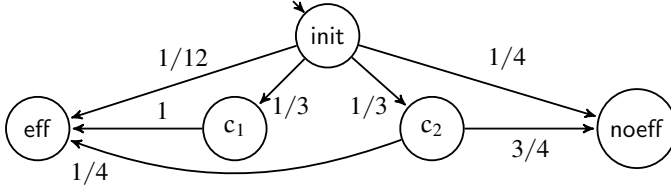
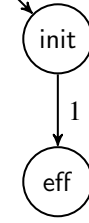Figure 1: A MC allowing for non-strict GPR causes

Figure 2: A MC in which no PR cause exists

Example 3.4 (Non-strict GPR cause). Consider the Markov chain $\mathcal{M}$ depicted in Figure 1 where the nodes represent states and the directed edges represent transitions labeled with their respective probabilities. Let $\mathsf{Eff} = \{\mathsf{eff}\}$. Then,

$$\mathrm{Pr}_{\mathcal{M}}(\Diamond\mathsf{Eff}) = \tfrac{1}{3} + \tfrac{1}{3}\cdot\tfrac{1}{4} + \tfrac{1}{12} = \tfrac{1}{2}, \quad \mathrm{Pr}_{\mathcal{M}}(\Diamond\mathsf{Eff}|\Diamond c_1) = \mathrm{Pr}_{\mathcal{M},c_1}(\Diamond\mathsf{eff}) = 1 \text{ and}$$
$$\mathrm{Pr}_{\mathcal{M}}(\Diamond\mathsf{Eff}|\Diamond c_2) = \mathrm{Pr}_{\mathcal{M},c_2}(\Diamond\mathsf{eff}) = \tfrac{1}{4}.$$

Thus, $\{c_1\}$ is both an SPR and a GPR cause for Eff, while $\{c_2\}$ is not. The set $\mathsf{Cause} = \{c_1, c_2\}$ is a non-strict GPR cause for Eff as:

$$\mathrm{Pr}_{\mathcal{M}}(\ \Diamond\mathsf{Eff} \mid \Diamond\mathsf{Cause}\ ) = (\tfrac{1}{3} + \tfrac{1}{3}\cdot\tfrac{1}{4})/(\tfrac{1}{3} + \tfrac{1}{3}) = (\tfrac{5}{12})/(\tfrac{2}{3}) = \tfrac{5}{8} > \tfrac{1}{2} = \mathrm{Pr}_{\mathcal{M}}(\Diamond\mathsf{Eff}).$$

Non-strictness follows from the fact that the SPR condition does not hold for state c_2. ◁

Example 3.5 (Probability-raising causes might not exist). PR causes might not exist, even if $\mathcal{M}$ is a Markov chain. This applies, e.g., to the MC in Figure 2 and the effect set $\mathsf{Eff} = \{\mathsf{eff}\}$. The only cause candidate is the singleton $\{\mathsf{init}\}$. However, the strict inequality in (GPR) or (SPR) forbids $\{\mathsf{init}\}$ to be a PR cause. The same phenomenon occurs if all non-terminal states of a MC reach the effect states with the same probability. In such cases, however, the non-existence of PR causes is well justified as the events $\Diamond\mathsf{Eff}$ and $\Diamond\mathsf{Cause}$ are stochastically independent for every set $\mathsf{Cause} \subseteq S \setminus \mathsf{Eff}$. ◁

Remark 3.6 (Memory needed for refuting PR condition). Let $\mathcal{M}$ be the MDP in Figure 3, where the notation is similar to Example 3.4 with the addition of actions α, β and γ. Let $\mathsf{Cause} = \{c\}$ and $\mathsf{Eff} = \{\mathsf{eff}\}$. Only state s has a nondeterministic choice. Cause is not an PR cause. To see this, regard the finite-memory deterministic scheduler $\mathfrak{T}$ that schedules β only for the first visit of s and α for the second visit of s. Then:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{eff}) = \tfrac{1}{2}\cdot\tfrac{1}{2} + \tfrac{1}{2}\cdot\tfrac{1}{2}\cdot 1\cdot\tfrac{1}{4} = \tfrac{5}{16} > \tfrac{1}{4} = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{eff}|\Diamond c)$$
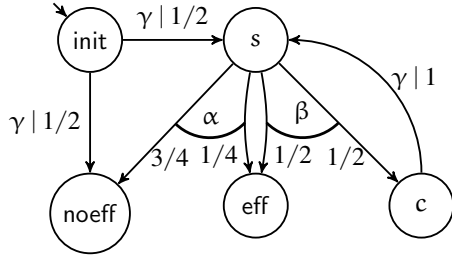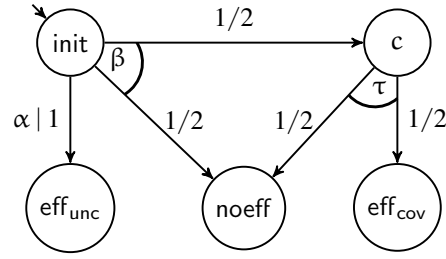
Denote the MR schedulers reaching c with positive probability as $\mathfrak{S}_\lambda$ with $\mathfrak{S}_\lambda(s)(\alpha) = \lambda$ and $\mathfrak{S}_\lambda(s)(\beta) = 1 - \lambda$ for some $\lambda \in [0, 1[$. Then, $\mathrm{Pr}_{\mathcal{M},s}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}) > 0$ and:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}) = \tfrac{1}{2}\cdot\mathrm{Pr}_{\mathcal{M},s}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}) < \mathrm{Pr}_{\mathcal{M},s}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}) = \mathrm{Pr}_{\mathcal{M},c}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_\lambda}(\Diamond\mathsf{eff}|\Diamond c)$$

Thus, the SPR/GPR condition holds for Cause and Eff under all memoryless schedulers reaching Cause with positive probability, although Cause is not an PR cause. ◁

Remark 3.7 (Randomization needed for refuting PR condition). Consider the MDP $\mathcal{M}$ in Figure 4. Let $\mathsf{Eff} = \{\mathsf{eff}_{\mathsf{unc}}, \mathsf{eff}_{\mathsf{cov}}\}$ and $\mathsf{Cause} = \{c\}$. Here the state $\mathsf{eff}_{\mathsf{unc}}$ is not covered by the cause whereas $\mathsf{eff}_{\mathsf{cov}}$ is, hence their names. The two MD-schedulers $\mathfrak{S}_\alpha$ and $\mathfrak{S}_\beta$ which select α resp. β for the initial state init are the only deterministic schedulers. As $\mathfrak{S}_\alpha$ does not reach c, it is irrelevant for the PR conditions. On the other hand $\mathfrak{S}_\beta$ satisfies (SPR) and (GPR) since

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_\beta}(\Diamond\mathsf{Eff}|\Diamond c) = \tfrac{1}{2} > \tfrac{1}{4} = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_\beta}(\Diamond\mathsf{Eff}).$$

Figure 3: MDP $\mathcal{M}$ from Remark 3.6   Figure 4: MDP $\mathcal{M}$ from Remark 3.7

The MR scheduler $\mathfrak{T}$ which selects α and β with probability $\frac{1}{2}$ in init also reaches c with positive probability but violates (SPR) and (GPR) as

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{Eff}|\Diamond\mathsf{c}) = \tfrac{1}{2} < \tfrac{5}{8} = \tfrac{1}{2} + \tfrac{1}{2} \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{Eff}). \qquad \triangleleft$$

Remark 3.8 (Cause-effect relations for regular classes of schedulers). The definitions of PR causes in MDPs impose constraints for all schedulers reaching a cause state. This condition is fairly strong and can often lead to the phenomenon that no PR cause exists. Replacing $\mathcal{M}$ with an MDP resulting from the synchronous parallel composition of $\mathcal{M}$ with a deterministic finite automaton representing a regular constraint on the scheduled state-action sequences (e.g., "alternate between actions α and β in state s" or "take α on every third visit to state s and actions β or γ otherwise") leads to a weaker notion of PR causality. This can be useful to obtain more detailed information on cause-effect relationships in special scenarios, be it at design time where multiple scenarios (regular classes of schedulers) are considered or for a post-hoc analysis where one seeks for the causes of an occurred effect and where information about the scheduled actions is extractable from log files or the information gathered by a monitor. $\triangleleft$

Remark 3.9 (Action PR causality). Our notions of PR causes are purely state-based with PR conditions that compare probabilities under the same scheduler. However, in combination with model transformations, the proposed notions of PR causes are also applicable for reasoning about other forms of PR causality.

Suppose, the task is to check whether taking action α in state s raises the effect probabilities compared to never scheduling α in state s. This form of action causality was discussed in an example in [DFT20]. We argue that we can deal with this kind of causality to. For this we assume there are no cycles in $\mathcal{M}$ containing s. Let $\mathcal{M}_0$ and $\mathcal{M}_1$ be copies of $\mathcal{M}$ with the following modifications: In $\mathcal{M}_0$, the only enabled action of state s is α, while in $\mathcal{M}_1$ the enabled actions of state s are the elements of $Act_{\mathcal{M}}(\mathsf{s}) \setminus \{\alpha\}$.

The action α raises the effect probability in $\mathcal{M}$ if and only if for all scheduler $\mathfrak{S}$ of $\mathcal{N}$ the copy of s in $\mathcal{M}_0$ satisfies (SPR) in $\mathcal{N}$. This idea can be generalized to check whether scheduler classes satisfying a regular constraint have higher effect probability compared to all other schedulers. In this case, we can deal with an MDP $\mathcal{N}$ as above where $\mathcal{M}_0$ and $\mathcal{M}_1$ are defined as the synchronous product of deterministic finite automata and $\mathcal{M}$.

To demonstrate this consider the MDP $\mathcal{M}$ from Figure 5. We are interested whether taking α in s raises the probability to reach the effect state eff. The constructed MDP $\mathcal{N}$ with two adapted copies of $\mathcal{M}$ is depicted in Figure 6. For all scheduler $\mathfrak{S}$ of $\mathcal{N}$ the state s_0 satisfies (SPR) by

$$\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond\mathsf{eff} \mid \Diamond\mathsf{s}_0) = 1/4 > 1/8 = \mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond\mathsf{eff}),$$

which means that the action α does indeed raise the probability of eff in $\mathcal{M}$. $\triangleleft$
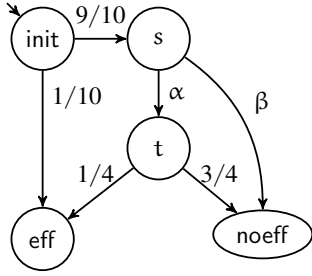
Figure 5: MDP $\mathcal{M}$ from Remark 3.9
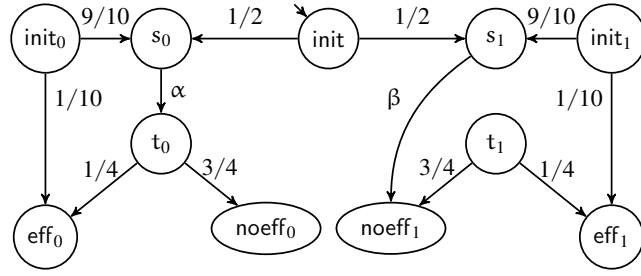
Figure 6: $\mathcal{N}$ with the two copies $\mathcal{M}_0$ and $\mathcal{M}_1$

4. CHECKING THE EXISTENCE OF PR CAUSES AND THE PR CONDITIONS

We now turn to algorithms for checking whether a given set Cause is an SPR or GPR cause for Eff in $\mathcal{M}$. Since the minimality condition (for all $c \in$ Cause $: \text{Pr}_{\mathcal{M}}^{\max}(\neg \text{Cause} \cup c) > 0$) of PR causes is verifiable by standard model checking techniques in polynomial time, we concentrate on checking the probability-raising conditions (S) and (G). In the special case where $\mathcal{M}$ is a Markov chain, both conditions (SPR) and (GPR) can be checked in polynomial time by computing the corresponding probabilities. Thus, the interesting case is checking the PR conditions in MDPs. In case of SPR causality this is closely related to the existence of PR causes and decidable in polynomial time (Section 4.1), while checking the GPR condition is more complex and polynomially reducible to (the non-solvability of) a quadratic constraint system (Section 4.2).

We start with the preliminary consideration that for both conditions (S) and (G), it suffices to consider a class of worst-case scheduler, which are minimizing the probability to reach an effect state from every cause state. For this we transform the MDP in question according to the cause candidate in question.

Notation 4.1 (MDP with minimal effect probabilities from cause candidates). If $C \subseteq S$ then we write $\mathcal{M}_{[C]}$ for the MDP resulting from $\mathcal{M}$ by removing all enabled actions and transitions of the states in C. Instead, $\mathcal{M}_{[C]}$ has a fresh action γ which is enabled exactly in the states $s \in C$ with the transition probabilities $P_{\mathcal{M}_{[C]}}(s, \gamma, \text{eff}) = \text{Pr}_{\mathcal{M},s}^{\min}(\Diamond \text{Eff})$ and $P_{\mathcal{M}_{[C]}}(s, \gamma, \text{noeff}) = 1 - \text{Pr}_{\mathcal{M},s}^{\min}(\Diamond \text{Eff})$. Here, eff is a fixed state in Eff and noeff a (possibly fresh) terminal state not in Eff. We write $\mathcal{M}_{[c]}$ if $C = \{c\}$ is a singleton. ◁

As an example for the model transformation consider the abstract MDP $\mathcal{M}$ from Figure 7 for the singleton set $C = \{c\}$. The transformed MDP $\mathcal{M}_{[c]}$ is seen in Figure 8, where a fresh state noeff is added. Furthermore, all outgoing transitions from c are deleted a replaced by a fresh action γ with exactly two transitions corresponding to $P_{\mathcal{M}_{[c]}}(c, \gamma, \text{eff}) = \text{Pr}_{\mathcal{M},c}^{\min}(\Diamond \text{Eff})$ and $P_{\mathcal{M}_{[c]}}(c, \gamma, \text{noeff}) = 1 - \text{Pr}_{\mathcal{M},c}^{\min}(\Diamond \text{Eff})$.

Lemma 4.2. *Let* $\mathcal{M} = (S, Act, P, \text{init})$ *be an MDP and* Eff $\subseteq S$ *a set of terminal states. Let* Cause $\subseteq S \setminus$ Eff. *Then,* Cause *is an SPR cause (resp. a GPR cause) for* Eff *in* $\mathcal{M}$ *if and only if* Cause *is an SPR cause (resp. a GPR cause) for* Eff *in* $\mathcal{M}_{[\text{Cause}]}$.

Obviously, for all $c \in$ Cause $: \text{Pr}_{\mathcal{M}}^{\max}(\neg \text{Cause} \cup c) > 0$ holds for Cause in $\mathcal{M}$ if and only if it holds for Cause in $\mathcal{M}_{[\text{Cause}]}$. Furthermore, it is clear all SPR resp. GPR causes of $\mathcal{M}$ are also SPR resp. GPR causes in $\mathcal{M}_{[\text{Cause}]}$. So, it remains to prove the converse direction. This will be done in Lemma 4.3 for SPR causes and in Lemma 4.4 for GPR causes.
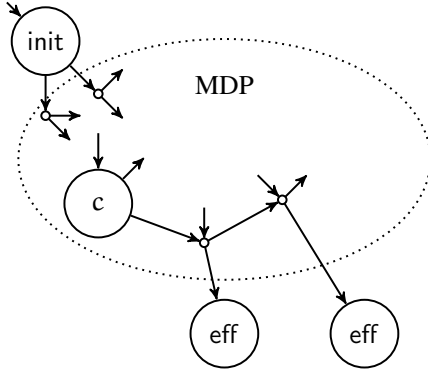
Figure 7: MDP $\mathcal{M}$
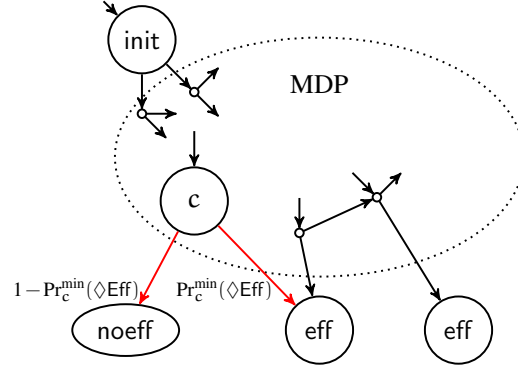
Figure 8: Transformed MDP $\mathcal{M}_{[\mathsf{c}]}$

Lemma 4.3 (Criterion for strict probability-raising causes). *Suppose* Cause *is an SPR cause for* Eff *in* $\mathcal{M}_{[\mathsf{Cause}]}$. *Then,* Cause *is an SPR cause for* Eff *in* $\mathcal{M}$.

Proof. We show that Cause is an SPR cause in $\mathcal{M}$ by showing (S) for all states in Cause. Thus, we fix a state $\mathsf{c} \in$ Cause. Recall also that we assume the states in Eff to be terminal. Let $\psi_{\mathsf{c}} = (\neg\mathsf{Cause})\, U\, \mathsf{c}$, $w_{\mathsf{c}} = \mathrm{Pr}^{\min}_{\mathcal{M},\mathsf{c}}(\lozenge\mathsf{Eff})$ and let Υ_{c} denote the set of all schedulers $\mathfrak{U}$ for $\mathcal{M}$ such that

- $\mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}(\psi_{\mathsf{c}}) > 0$ and
- $\mathrm{Pr}^{res(\mathfrak{U},\pi)}_{\mathcal{M},\mathsf{c}}(\lozenge\mathsf{Eff}) = w_{\mathsf{c}}$ for each finite $\mathfrak{U}$-path π from init to c.

Clearly, $\mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}(\lozenge\mathsf{c} \wedge \lozenge\mathsf{Eff}) = \mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}(\lozenge\mathsf{c}) \cdot w_{\mathsf{c}}$ for $\mathfrak{U} \in \Upsilon_{\mathsf{c}}$. As Cause is an SPR cause in $\mathcal{M}_{[\mathsf{Cause}]}$ we have:

$$w_{\mathsf{c}} \;>\; \mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) \quad \text{for all schedulers } \mathfrak{U} \in \Upsilon_{\mathsf{c}}. \tag{4.1}$$

The task is to prove that (S) holds for c and all schedulers of $\mathcal{M}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}}) > 0$.

Suppose $\mathfrak{S}$ is a scheduler for $\mathcal{M}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}}) > 0$. Then $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}} \wedge \lozenge\mathsf{Eff}) \geqslant \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}}) \cdot w_{\mathsf{c}}$. Moreover, there exists a scheduler $\mathfrak{U} = \mathfrak{U}_{\mathfrak{S}} \in \Upsilon_{\mathsf{c}}$ with

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}}) = \mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}(\psi_{\mathsf{c}}) \quad \text{and} \quad \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\psi_{\mathsf{c}}) \wedge \lozenge\mathsf{Eff}) = \mathrm{Pr}^{\mathfrak{U}}_{\mathcal{M}}((\neg\psi_{\mathsf{c}}) \wedge \lozenge\mathsf{Eff}).$$

To see this, consider the scheduler $\mathfrak{U}$ that behaves as $\mathfrak{S}$ as long as c is not reached. As soon as $\mathfrak{U}$ has reached c, scheduler $\mathfrak{U}$ switches mode and behaves as an MD-scheduler minimizing the probability to reach an effect state. The SPR condition (S) holds for c and $\mathfrak{S}$ if and only if

$$\frac{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}} \wedge \lozenge\mathsf{Eff})}{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}})} \;>\; \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) \tag{4.2}$$

Using

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) \;=\; \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}} \wedge \lozenge\mathsf{Eff}) + \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\psi_{\mathsf{c}}) \wedge \lozenge\mathsf{Eff}),$$

we can equivalently convert condition (4.2) for c and $\mathfrak{S}$ to

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}} \wedge \lozenge\mathsf{Eff}) \cdot \frac{1 - \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}})}{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\psi_{\mathsf{c}})} \;>\; \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\psi_{\mathsf{c}}) \wedge \lozenge\mathsf{Eff}) \tag{4.3}$$

The remaining task is now to derive (4.3) from (4.1). Applying (4.1) to scheduler $\mathfrak{U} = \mathfrak{U}_{\mathfrak{S}}$ yields:

$$
\begin{aligned}
w_c \quad &> \quad \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\psi_c \wedge \Diamond \mathsf{Eff}) + \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}((\neg \psi_c) \wedge \Diamond \mathsf{Eff}) \\
&= \quad \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c) \cdot w_c + \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \psi_c) \wedge \Diamond \mathsf{Eff}).
\end{aligned}
$$

We conclude:

$$
\begin{aligned}
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c \wedge \Diamond \mathsf{Eff}) \cdot \frac{1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c)}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c)} \quad &\geqslant \quad \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c) \cdot w_c \cdot \frac{1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c)}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c)} \\
&= \quad \big(1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\psi_c)\big) \cdot w_c \\
&> \quad \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \psi_c) \wedge \Diamond \mathsf{Eff}).
\end{aligned}
$$

Thus, (4.3) holds for c and $\mathfrak{S}$. $\qquad\square$

Lemma 4.4 (Criterion for GPR causes). *Suppose* Cause *is a GPR cause for* Eff *in* $\mathcal{M}_{[\mathsf{Cause}]}$. *Then,* Cause *is a GPR cause for* Eff *in* $\mathcal{M}$.

Proof. From the assumption that Cause is a GPR cause for Eff in $\mathcal{M}_{[\mathsf{Cause}]}$, we can conclude that the GPR condition (GPR) holds for all schedulers $\mathfrak{S}$ that satisfy

$$
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) > 0 \qquad \text{and} \qquad \mathrm{Pr}_{\mathcal{M},c}^{res(\mathfrak{S},\pi)}(\Diamond \mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M},c}^{\min}(\Diamond \mathsf{Eff})
$$

for each finite $\mathfrak{S}$-path from the initial state init to a state $c \in \mathsf{Cause}$. To prove that (GPR) holds for all schedulers $\mathfrak{S}$ that satisfy $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) > 0$, we introduce the following notation: We write

- $\Sigma_{>0}$ for the set of all schedulers $\mathfrak{S}$ such that $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) > 0$,
- $\Sigma_{>0,\min}$ for the set of all schedulers with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) > 0$ such that

$$
\mathrm{Pr}_{\mathcal{M},c}^{res(\mathfrak{S},\pi)}(\Diamond \mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M},c}^{\min}(\Diamond \mathsf{Eff})
$$

for each finite $\mathfrak{S}$-path from the initial state init to a state $c \in \mathsf{Cause}$.

It now suffices to show that for each scheduler $\mathfrak{S} \in \Sigma_{>0}$ there exists a scheduler $\mathfrak{S}' \in \Sigma_{>0,\min}$ such that if (GPR) holds for $\mathfrak{S}'$ then (GPR) holds for $\mathfrak{S}$. So, let $\mathfrak{S} \in \Sigma_{>0}$.

For $c \in \mathsf{Cause}$, let Π_c denote the set of finite paths $\pi = s_0 \, \alpha_0 \, s_1 \, \alpha_1 \dots \alpha_{n-1} \, s_n$ with $s_0 = \mathsf{init}$, $s_n = c$ and $\{s_0, \dots, s_{n-1}\} \cap (\mathsf{Cause} \cup \mathsf{Eff}) = \varnothing$. Let

$$
w_\pi^{\mathfrak{S}} = \mathrm{Pr}_{\mathcal{M},c}^{res(\mathfrak{S},\pi)}(\Diamond \mathsf{Eff})
$$

Furthermore, let $p_\pi^{\mathfrak{S}}$ denote the probability for (the cylinder set of) π under scheduler $\mathfrak{S}$. Then

$$
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg \mathsf{Cause}) \, \mathsf{U} \, c) = \sum_{\pi \in \Pi_c} p_\pi^{\mathfrak{S}}.
$$

Moreover:

$$
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg \mathsf{Cause} \, \mathsf{U} \, \mathsf{Eff}) + \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_\pi^{\mathfrak{S}} \cdot w_\pi^{\mathfrak{S}} \text{ and}
$$

$$
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff} \mid \Diamond \mathsf{Cause}) = \frac{1}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause})} \cdot \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_\pi^{\mathfrak{S}} \cdot w_\pi^{\mathfrak{S}}
$$

Thus, the condition (GPR) holds for the scheduler $\mathfrak{S} \in \Sigma_{>0}$ if and only if

$$
\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg \mathsf{Cause} \, \mathsf{U} \, \mathsf{Eff}) + \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_\pi^{\mathfrak{S}} \cdot w_\pi^{\mathfrak{S}} \quad < \quad \frac{1}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause})} \cdot \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_\pi^{\mathfrak{S}} \cdot w_\pi^{\mathfrak{S}}.
$$

The latter is equivalent to:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge\mathsf{Cause}) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg\mathsf{Cause}\,\mathsf{U}\,\mathsf{Eff}) + \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge\mathsf{Cause}) \cdot \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_{\pi}^{\mathfrak{S}} \cdot w_{\pi}^{\mathfrak{S}} < \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_{\pi}^{\mathfrak{S}} \cdot w_{\pi}^{\mathfrak{S}},$$

which again is equivalent to:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge\mathsf{Cause}) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg\mathsf{Cause}\,\mathsf{U}\,\mathsf{Eff}) < \big(1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge\mathsf{Cause})\big) \cdot \sum_{c \in \mathsf{Cause}} \sum_{\pi \in \Pi_c} p_{\pi}^{\mathfrak{S}} \cdot w_{\pi}^{\mathfrak{S}}. \qquad (4.4)$$

Pick an MD-scheduler $\mathfrak{T}$ that minimizes the probability to reach Eff from every state. In particular, $w_c = w_{\pi}^{\mathfrak{T}} \leqslant w_{\pi}^{\mathfrak{S}}$ for every state $c \in \mathsf{Cause}$ and every path $\pi \in \Pi_c$ (recall that $w_c = \mathrm{Pr}_{\mathcal{M},c}^{\min}(\lozenge\mathsf{Eff})$). Moreover, the scheduler $\mathfrak{S}$ can be transformed into a scheduler $\mathfrak{S}_{\mathfrak{T}} \in \Sigma_{>0,\min}$ that is "equivalent" to $\mathfrak{S}$ with respect to the global probability-raising condition. More concretely, let $\mathfrak{S}_{\mathfrak{T}}$ denote the scheduler that behaves as $\mathfrak{S}$ as long as $\mathfrak{S}$ has not yet visited a state in Cause and behaves as $\mathfrak{T}$ as soon as a state in Cause has been reached. Thus, $p_{\pi}^{\mathfrak{S}} = p_{\pi}^{\mathfrak{S}_{\mathfrak{T}}}$ and $res(\mathfrak{S}_{\mathfrak{T}}, \pi) = \mathfrak{T}$ for each $\pi \in \Pi_c$. This yields that the probability to reach $c \in \mathsf{Cause}$ from init is the same under $\mathfrak{S}$ and $\mathfrak{S}_{\mathfrak{T}}$, i.e., $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge c) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_{\mathfrak{T}}}(\lozenge c)$. Therefore $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge\mathsf{Cause}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_{\mathfrak{T}}}(\lozenge\mathsf{Cause})$. The latter implies that $\mathfrak{S}_{\mathfrak{T}} \in \Sigma_{>0}$, and hence $\mathfrak{S}_{\mathfrak{T}} \in \Sigma_{>0,\min}$. Moreover, $\mathfrak{S}$ and $\mathfrak{S}_{\mathfrak{T}}$ reach Eff without visiting Cause with the same probability, i.e., $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg\mathsf{Cause}\,\mathsf{U}\,\mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_{\mathfrak{T}}}(\neg\mathsf{Cause}\,\mathsf{U}\,\mathsf{Eff})$.

But this yields: if (4.4) holds for $\mathfrak{S}_{\mathfrak{T}}$ then (4.4) holds for $\mathfrak{S}$. As (4.4) holds for $\mathfrak{S}_{\mathfrak{T}}$ by assumption, this completes the proof. □

4.1. **Checking the strict probability-raising condition and the existence of causes.** The basis of both checking the existence of PR causes or checking the SPR condition (S) for a given cause candidate is the following polynomial time algorithm to check whether (S) holds in a given state c of $\mathcal{M}$ for all schedulers $\mathfrak{S}$ with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\lozenge c) > 0$:
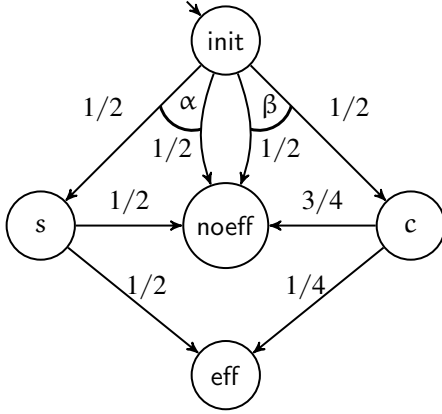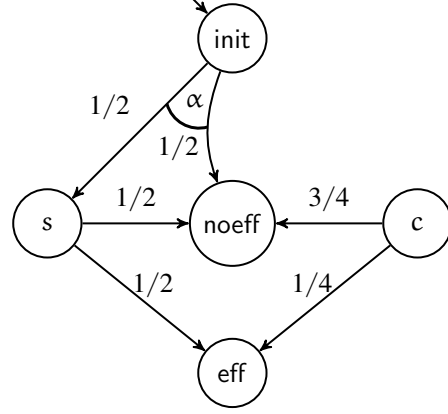
Algorithm 4.5.

Input: state $c \in S$, set of terminal states $\mathsf{Eff} \subseteq S$
Task: Decide whether (SPR) holds in c for all schedulers $\mathfrak{S}$.

0. Compute $q_s = \mathrm{Pr}_{\mathcal{M}_{[c]},s}^{\max}(\lozenge\mathsf{Eff})$ and $w_c = \mathrm{Pr}_{\mathcal{M},c}^{\min}(\lozenge\mathsf{Eff})$ for each state s in $\mathcal{M}_{[c]}$.
1. If $q_{\mathsf{init}} < w_c$, then return "yes, (SPR) holds for c".
2. If $q_{\mathsf{init}} > w_c$, then return "no, (SPR) does not hold for c".
3. Suppose $q_{\mathsf{init}} = w_c$. Let $A(s) = \{\alpha \in Act_{\mathcal{M}_{[c]}}(s) \mid q_s = \sum_{t \in S_{[c]}} P_{\mathcal{M}_{[c]}}(s, \alpha, t) \cdot q_t\}$ for each non-terminal state s. Let $\mathcal{M}_{[c]}^{\max}$ denote the sub-MDP of $\mathcal{M}_{[c]}$ induced by the state-action pairs (s, α) where $\alpha \in A(s)$.
   3.1 If c is reachable from init in $\mathcal{M}_{[c]}^{\max}$, then return "no, (SPR) does not hold for c".
   3.2 If c is not reachable from init in $\mathcal{M}_{[c]}^{\max}$, then return "yes, (SPR) holds for c".

As the construction of the MDP $\mathcal{M}_{[c]}$ suggests, the two values compared by the algorithm are instances of worst-case scheduler. On one hand, the probability to reach Eff starting in c is minimized, while it is maximized if c was not seen yet. If in such a scenario we have case 1. $q_{\mathsf{init}} < w_c$ then c obviously satisfies (SPR). In the case 2. $q_{\mathsf{init}} > w_c$ we can build a scheduler which refuses (SPR) for c. Lastly, in the corner case 3. $q_{\mathsf{init}} = w_c$ a treatment by a reachability analysis is needed, as seen in the following Example 4.6.

Figure 9: Example MDP $\mathcal{M}_{[c]}$

Figure 10: Transformed MDP $\mathcal{M}_{[c]}^{\max}$

Example 4.6. For the transformation to $\mathcal{M}_{[c]}^{\max}$ consider $\mathcal{M}_{[c]}$ from Figure 9. For Cause $= \{c\}$ we are in case 3. of Algorithm 4.5 as $q_{\mathsf{init}} = \mathrm{Pr}_{\mathcal{M}_{[c]},s}^{\max}(\Diamond \mathsf{Eff}) = 1/4 = \mathrm{Pr}_{\mathcal{M},c}^{\min}(\Diamond \mathsf{Eff}) = w_c$. The only non-deterministic choice is in the state init. We have $A(\mathsf{init}) = \{\alpha\}$ since α is the only maximizing action for $\Diamond \mathsf{eff}$ in init. Thus, in the resulting MDP $\mathcal{M}_{[c]}^{\max}$, depicted in Figure 10, all other actions in init are deleted. We are actually in case 3.2 as c is not reachable from the initial state in $\mathcal{M}_{[c]}^{\max}$.   ◁

Lemma 4.7. *Algorithm 4.5 is sound and runs in polynomial time.*

Proof. First, we show the soundness of Algorithm 4.5. By the virtue of Lemma 4.2 stating the soundness of the tranformation $\mathcal{M}$ to $\mathcal{M}_{[c]}$ it suffices to show that Algorithm 4.5 returns the correct answers "yes" or "no" when the task is to check whether the singleton Cause $= \{c\}$ is an SPR cause in $\mathcal{N} = \mathcal{M}_{[c]}$. Recall the notation $q_s = \mathrm{Pr}_{\mathcal{M}_{[c]},s}^{\max}(\Diamond \mathsf{Eff})$. We abbreviate $q = q_{\mathsf{init}}$. Note that $(\neg \mathsf{Cause})\,U\,c$ is equivalent to $\Diamond c$ as $c \in \mathsf{Cause}$.

For every scheduler $\mathfrak{S}$ of $\mathcal{N}$ we have $\mathrm{Pr}_{\mathcal{N},c}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) = w_c$. Thus, $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond \mathsf{Eff} \mid \Diamond c) = w_c$ if $\mathfrak{S}$ is a scheduler of $\mathcal{N}$ with $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond c) > 0$.

Algorithm 4.5 correctly answers "no" (case 2 or 3.1) if $w_c = 0$. Suppose that $w_c > 0$. Thus, the SPR condition for c reduces to $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) < w_c$ for all schedulers $\mathfrak{S}$ of $\mathcal{N}$ with $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond c) > 0$.

1. of Algorithm 4.5 (i.e., if $q < w_c$), the answer "yes" is sound as then $\mathrm{Pr}_{\mathcal{N}}^{\max}(\Diamond \mathsf{Eff}) = q < w_c$.

2. (i.e., if $q > w_c$) Let $\mathfrak{T}$ be an MD-scheduler with $\mathrm{Pr}_{\mathcal{N},s}^{\mathfrak{T}}(\Diamond \mathsf{Eff}) = q_s$ for each state s and pick an MD-scheduler $\mathfrak{S}$ with $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond c) > 0$. It is no restriction to suppose that $\mathfrak{T}$ and $\mathfrak{S}$ realize the same end components of $\mathcal{N}$. (Note that if state s belongs to an end component that is realized by $\mathfrak{T}$ then s contained in a bottom strongly connected component of the Markov chain induced by $\mathfrak{T}$. But then $q_s = 0$, i.e., no effect state is reachable from s in $\mathcal{N}$. Recall that all effect states are terminal and thus not contained in end components. But then we can safely assume that $\mathfrak{T}$ and $\mathfrak{S}$ schedule the same action for state s.) Let λ be any real number with $1 > \lambda > \frac{w_c}{q}$ and let $\mathcal{K}$ denote the sub-MDP of $\mathcal{N}$ with state space S where the enabled actions of state s are the actions scheduled for s under one of the schedulers $\mathfrak{T}$ or $\mathfrak{S}$. Let now $\mathfrak{U}$ be the MR-scheduler $\lambda \mathfrak{T} \oplus (1-\lambda)\mathfrak{S}$ defined as in Lemma 2.2 for the EC-free MDP resulting from $\mathcal{K}$ when collapsing $\mathcal{K}$'s end components into a single terminal state. For the states belonging to an end component

of $\mathcal{K}$, $\mathfrak{U}$ schedules the same action as $\mathfrak{T}$ and $\mathfrak{S}$. Then, $\Pr_{\mathcal{N}}^{\mathfrak{U}}(\Diamond t) = \lambda \Pr_{\mathcal{N}}^{\mathfrak{T}}(\Diamond t) + (1-\lambda) \Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond t)$ for all terminal states t of $\mathcal{N}$ and $t = c$. Hence:

$$\Pr_{\mathcal{N}}^{\mathfrak{U}}(\Diamond c) \;\geqslant\; (1-\lambda) \cdot \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond c) \;>\; 0 \text{ and}$$

$$\Pr_{\mathcal{N}}^{\mathfrak{U}}(\Diamond \mathsf{Eff}) \;\geqslant\; \lambda \cdot \Pr_{\mathcal{M}}^{\mathfrak{T}}(\Diamond \mathsf{Eff}) \;=\; \lambda \cdot q \;>\; w_c$$

Thus, scheduler $\mathfrak{U}$ is a witness why (SPR) does not hold for c.

3.1 Pick an MD-scheduler $\mathfrak{S}$ of $\mathcal{M}_{[c]}^{\max}$ such that c is reachable from init via $\mathfrak{S}$ and $\Pr_{\mathcal{N},s}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) = q_s$ for all states s. Hence, (SPR) does not hold for c and the scheduler $\mathfrak{S}$.

3.2 We have the property that $\Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond c) = 0$ for all schedulers $\mathfrak{S}$ for $\mathcal{N}$ with $\Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) = q = w_c$. But then $\Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond c) > 0$ implies $\Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) < w_c$ as required in (SPR).

The polynomial runtime of Algorithm 4.5 follows from the fact that minimal and maximal reachability probabilities and hence also the MDPs $\mathcal{N} = \mathcal{M}_{[c]}$ and its sub-MDP $\mathcal{M}_{[c]}^{\max}$ can be computed in polynomial time. $\qquad\square$

By applying Algorithm 4.5 to all states $c \in \mathsf{Cause}$ and standard algorithms to check the existence of a path satisfying $(\neg \mathsf{Cause}) \, \mathsf{U} \, c$ for every state $c \in \mathsf{Cause}$, we obtain:

Theorem 4.8 (Checking SPR causes). *The problem "given $\mathcal{M}$, Cause and Eff, check whether Cause is a SPR cause for Eff in $\mathcal{M}$" is solvable in polynomial-time.*

Remark 4.9 (Memory requirements for (S)). As the soundness proof for Algorithm 4.5 shows: If Cause does not satisfy (S), then there is an MR-scheduler $\mathfrak{S}$ for $\mathcal{M}_{[\mathsf{Cause}]}$ witnessing the violation of (SPR). Scheduler $\mathfrak{S}$ corresponds to a finite-memory (randomized) scheduler $\mathfrak{T}$ with two memory cells for $\mathcal{M}$: "before Cause" (where $\mathfrak{T}$ behaves as $\mathfrak{S}$) and "after Cause" (where $\mathfrak{T}$ behaves as an MD-scheduler minimizing the effect probability). $\qquad\triangleleft$

Lemma 4.10 (Criterion for the existence of PR causes). *Let $\mathcal{M}$ be an MDP and Eff a nonempty set of states. The following statements are equivalent:*

(a) Eff *has an SPR cause in $\mathcal{M}$,*

(b) Eff *has a GPR cause in $\mathcal{M}$,*

(c) *there is a state $c_0 \in S \setminus \mathsf{Eff}$ such that the singleton $\{c_0\}$ is an SPR cause (and therefore a GRP cause) for Eff in $\mathcal{M}$.*

Thus, the existence of SPR and GPR causes can be checked with Algorithm 4.5 in polynomial time.

Proof. Obviously, statement (c) implies statements (a) and (b). The implication "(a) $\implies$ (b)" follows from Lemma 3.3. We now turn to the proof of "(b) $\implies$ (c)". For this, we assume that we are given a GPR cause Cause for Eff in $\mathcal{M}$. For $c \in \mathsf{Cause}$, let $w_c = \Pr_{\mathcal{M},c}^{\min}(\Diamond \mathsf{Eff})$. Pick a state $c_0 \in \mathsf{Cause}$ such that $w_{c_0} = \max\{w_c : c \in \mathsf{Cause}\}$. For every scheduler $\mathfrak{S}$ for $\mathcal{M}$ that minimizes the effect probability whenever it visits a state in Cause, and visits Cause with positive probability, the conditional probability $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff} | \Diamond \mathsf{Cause})$ is a weighted average of the values w_c, $c \in \mathsf{Cause}$, and thus bounded by w_{c_0}. Using Lemma 4.2 we see that it is sufficient to only consider the minimal probabilities $w_c = \Pr_{\mathcal{M},c}^{\min}(\Diamond \mathsf{Eff})$. Thus, we conclude that $\{c_0\}$ is both an SPR and a GPR cause for Eff. $\qquad\square$

4.2. **Checking the global probability-raising condition.** Throughout this section, we suppose that both the effect set Eff and the cause candidate Cause are fixed disjoint subsets of the state space of the MDP $\mathcal{M} = (S, Act, P, \text{init})$, and address the task to check whether Cause is a global probability-raising cause for Eff in $\mathcal{M}$. As the minimality condition (for all $c \in \mathsf{Cause} : \Pr_{\mathcal{M}}^{\max}(\neg \mathsf{Cause} \, \mathsf{U} \, c) > 0$) can be checked in polynomial time using a standard graph algorithm, we will concentrate on an algorithm to check the probability-raising condition (GPR). We start by stating the main results of this section.

Theorem 4.11. *Given* $\mathcal{M}$*,* Cause *and* Eff*, deciding whether* Cause *is a GPR cause for* Eff *in* $\mathcal{M}$ *can be done in* coNP.

In order to provide an algorithm, we perform a model transformation after which the violation of (GPR) by a scheduler $\mathfrak{S}$ can be expressed solely in terms of the expected frequencies of the state-action pairs of the transformed MDP under $\mathfrak{S}$. This allows us to express the existence of a scheduler witnessing the non-causality of Cause in terms of the satisfiability of a quadratic constraint system. Thus, we can restrict the quantification in (G) to MR-schedulers in the transformed model. We trace back the memory requirements to $\mathcal{M}_{[\mathsf{Cause}]}$ and to the original MDP $\mathcal{M}$ yielding the second main result.

Theorem 4.12. *Let* $\mathcal{M}$ *be an MDP with effect set* Eff *as before and* Cause *a set of non-effect states which satisfies for all* $\mathsf{c} \in \mathsf{Cause} : \mathrm{Pr}_{\mathcal{M}}^{\max}(\neg\mathsf{Cause}\,\mathrm{U}\,\mathsf{c}) > 0$*. If* Cause *is not a GPR cause for* Eff*, then there is an MR-scheduler for* $\mathcal{M}_{[\mathsf{Cause}]}$ *refuting the GPR condition for* Cause *in* $\mathcal{M}_{[\mathsf{Cause}]}$ *and a finite-memory scheduler for* $\mathcal{M}$ *with two memory cells refuting the GPR condition for* Cause *in* $\mathcal{M}$*.*

The remainder of this section is concerned with the proofs of both Theorem 4.11 and Theorem 4.12. For this, we suppose that Cause satisfies for all $\mathsf{c} \in \mathsf{Cause} : \mathrm{Pr}_{\mathcal{M}}^{\max}(\neg\mathsf{Cause}\,\mathrm{U}\,\mathsf{c}) > 0$ which can be checked preemptively in polynomial time as argued before.

Checking the GPR condition (Proof of Theorem 4.11). We will start with a polynomial-time model transformation into a kind of "canonical form" after which we can make the following assumptions when checking the GPR condition of Cause for Eff

(A1): $\mathsf{Eff} = \{\mathsf{eff}_{\mathsf{unc}}, \mathsf{eff}_{\mathsf{cov}}\}$ consists of two terminal states.

(A2): For every $\mathsf{c} \in \mathsf{Cause}$, there is a single enabled action $Act(\mathsf{c}) = \{\gamma\}$, and there is $w_{\mathsf{c}} \in [0,1] \cap \mathbb{Q}$ such that $P(\mathsf{c}, \gamma, \mathsf{eff}_{\mathsf{cov}}) = w_{\mathsf{c}}$ and $P(\mathsf{c}, \gamma, \mathsf{noeff}_{\mathsf{fp}}) = 1 - w_{\mathsf{c}}$, where $\mathsf{noeff}_{\mathsf{fp}}$ is a terminal non-effect state and $\mathsf{noeff}_{\mathsf{fp}}$ and $\mathsf{eff}_{\mathsf{cov}}$ are only accessible via γ-transition from the $\mathsf{c} \in \mathsf{Cause}$.

(A3): $\mathcal{M}$ has no end components and there is a further terminal state $\mathsf{noeff}_{\mathsf{tn}}$ and an action τ such that $\tau \in Act(s)$ implies $P(s, \tau, \mathsf{noeff}_{\mathsf{tn}}) = 1$.

The terminal states $\mathsf{eff}_{\mathsf{unc}}$, $\mathsf{eff}_{\mathsf{cov}}$, $\mathsf{noeff}_{\mathsf{fp}}$ and $\mathsf{noeff}_{\mathsf{tn}}$ are pairwise distinct. $\mathcal{M}$ can have further terminal states representing true negatives. However, these can be identified with $\mathsf{noeff}_{\mathsf{tn}}$.

Intuitively, $\mathsf{eff}_{\mathsf{cov}}$ stands for covered effects ("Eff after Cause") and can be seen as a true positive, while $\mathsf{eff}_{\mathsf{unc}}$ represents the uncovered effects ("Eff without preceding Cause") and corresponds to a false negative. Let $\mathfrak{S}$ be a scheduler in $\mathcal{M}$. Note that $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg\mathsf{Cause})\,\mathrm{U}\,\mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{unc}})$ and $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond(\mathsf{Cause} \wedge \Diamond\mathsf{Eff})) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{cov}})$. As the cause states can not reach each other we also have $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}((\neg\mathsf{Cause})\,\mathrm{U}\,\mathsf{c}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{c})$ for each $\mathsf{c} \in \mathsf{Cause}$. The intuitive meaning of $\mathsf{noeff}_{\mathsf{fp}}$ is a false positive ("no effect after Cause"), while $\mathsf{noeff}_{\mathsf{tn}}$ stands for true negatives where neither the effect nor the cause is observed. Note that $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond(\mathsf{Cause} \wedge \neg\Diamond\mathsf{Eff})) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{noeff}_{\mathsf{fp}})$ and $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\neg\Diamond\mathsf{Cause} \wedge \neg\Diamond\mathsf{Eff}) = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{noeff}_{\mathsf{tn}})$.

Establishing assumptions (A1)-(A3): We justify the assumptions as we can transform $\mathcal{M}$ into a new MDP of the same asymptotic size satisfying the above assumptions. Thanks to Lemma 4.2, we may suppose that $\mathcal{M} = \mathcal{M}_{[\mathsf{Cause}]}$ without changing the satisfaction of (G). Thus, from cause states $\mathsf{c} \in \mathsf{Cause}$ there are only two outgoing transitions, either to a terminal effect state eff with probability $\mathrm{Pr}_{\mathsf{c}}^{\min}(\Diamond\mathsf{Eff})$ or to a terminal non-effect state noeff with the remaining probability (see Notation 4.1). We then may rename the effect state eff and the non-effect state noeff reachable from Cause into $\mathsf{eff}_{\mathsf{cov}}$ and $\mathsf{noeff}_{\mathsf{fp}}$, respectively. Furthermore, we collapse all other effect states into a single state $\mathsf{eff}_{\mathsf{unc}}$ and all true negative states into $\mathsf{noeff}_{\mathsf{tn}}$. Similarly, by renaming and possibly duplicating terminal states we also suppose that $\mathsf{noeff}_{\mathsf{fp}}$ has no other incoming transitions than the
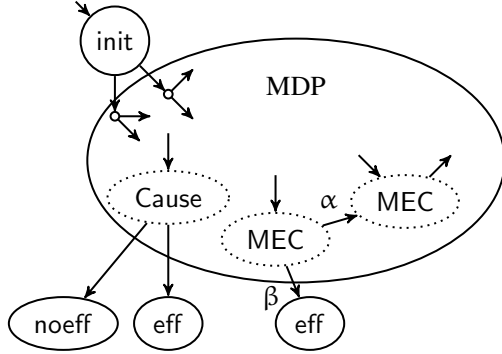
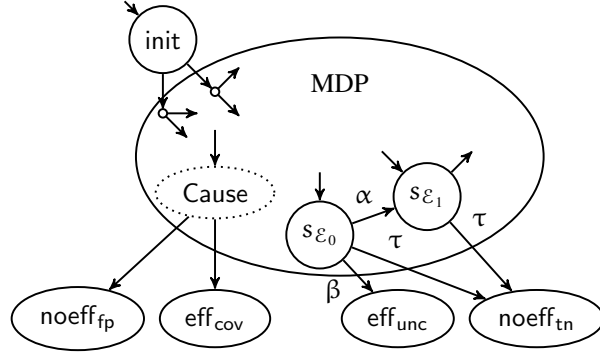Figure 11: MDP $\mathcal{M}$ not satisfying
assumptions (A1)-(A3)

Figure 12: Transformed MDP $\mathcal{N}$ satisfying
assumptions (A1)-(A3)

γ-transitions from the states in Cause. This ensures (A1) and (A2). For (A3) consider the set T of terminal states in the MDP obtained so far. We remove all non-trivial end components by switching to the MEC-quotient [dA97], i.e., we collapse all states that belong to the same MEC $\mathcal{E}$ into a single state $s_\mathcal{E}$ while ignoring the actions inside $\mathcal{E}$. Additionally, we add a fresh τ-transition from the states $s_\mathcal{E}$ to $\mathsf{noeff_{tn}}$ (i.e., $\mathsf{P}(s_\mathcal{E}, \tau, \mathsf{noeff_{tn}}) = 1$). The τ-transitions from states $s_\mathcal{E}$ to $\mathsf{noeff_{tn}}$ mimic cases where the scheduler of the original MDP enters the end component $\mathcal{E}$ and stays there forever.

In particular, consider the MEC-quotient $\mathcal{N}$ of $\mathcal{M}_{[\mathsf{Cause}]}$ (see Definition 2.3). Let $\mathsf{noeff_{tn}}$ be the state to which we add a τ-transition with probability 1 from each MEC that we collapse in the MEC-quotient. That is, $\mathsf{noeff_{tn}} = \bot$ with the notations of Definition 2.3.

We demonstrate these transformations on the abstract MDP $\mathcal{M}$ from Figure 11, where the dotted circles correspond to sets of states in the MDP. The MDP already satisfies $\mathcal{M} = \mathcal{M}_{[\mathsf{c}]}$. We rename eff reachable from Cause to $\mathsf{eff_{cov}}$ and noeff to $\mathsf{noeff_{fp}}$. Effect states not reachable from Cause collapse to $\mathsf{eff_{unc}}$. There are no terminal non-effect states not reachable from c, which would collapse to $\mathsf{noeff_{tn}}$. The MEC quotient collapses MECs to states $s_{\mathcal{E}_i}$ only keeping outgoing transitions. There is a fresh action τ in states $s_{\mathcal{E}_i}$ to $\mathsf{noeff_{tn}}$. Thus, we get $\mathcal{N}$ from Figure 12.

The following Lemma 4.13 and Corollary 4.14 prove the soundness of the model transformation.

Lemma 4.13. *For each scheduler $\mathfrak{S}$ for $\mathcal{M}_{[\mathsf{Cause}]}$, there is a scheduler $\mathfrak{T}$ for $\mathcal{N}$, and vice versa, such that*

- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\Diamond \mathsf{Eff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{N}}(\Diamond \mathsf{Eff})$,
- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\Diamond \mathsf{Cause}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{N}}(\Diamond \mathsf{Cause})$, *and*
- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\Diamond \mathsf{Cause} \wedge \Diamond \mathsf{Eff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{N}}(\Diamond \mathsf{eff_{cov}})$.

Proof. By Lemma 2.4, there is a scheduler $\mathfrak{T}$ for $\mathcal{N}$ for each scheduler $\mathfrak{S}$ for $\mathcal{M}_{[\mathsf{Cause}]}$ such that each terminal state is reached with the same probability under $\mathfrak{T}$ in $\mathcal{N}$ and under $\mathfrak{S}$ in $\mathcal{M}_{[\mathsf{Cause}]}$. The state $\mathsf{eff_{cov}}$ is present in $\mathcal{M}_{[\mathsf{Cause}]}$ under the name eff. The state eff is furthermore reached in $\mathcal{M}_{[\mathsf{Cause}]}$ if and only if $\Diamond \mathsf{Cause} \wedge \Diamond \mathsf{Eff}$ is satisfied along a run. The set of terminal states in Eff is obtained from the set Eff in $\mathcal{M}_{[\mathsf{Cause}]}$ by collapsing states. As a scheduler $\mathfrak{S}$ can be viewed as a scheduler for both MDPs and these MDPs agree except for the terminal states, the first equality follows as well. As the probability to reach Cause is the sum of the probabilities to reach the terminal states $\mathsf{eff_{cov}}$ and $\mathsf{noeff_{fp}}$ in $\mathcal{N}$ and as these states are only renamed in $\mathcal{N}$ in comparison to $\mathcal{M}_{[\mathsf{Cause}]}$, the claim follows. $\qquad\square$

From Lemma 4.13 and Lemma 4.2, we conclude the following corollary that justifies working under assumptions (A1)-(A3).

Corollary 4.14. *The set* Cause *is a GPR cause for* Eff *in* $\mathcal{M}$ *if and only if* Cause *is a GPR cause for* Eff *in* $\mathcal{N}$.

Proof. By Lemma 4.13, for each scheduler $\mathfrak{S}$ for $\mathcal{M}_{[\mathsf{Cause}]}$, there is a scheduler $\mathfrak{T}$ for $\mathcal{N}$ such that all relevant probabilities agree, and vice versa. So, Cause is a GPR cause for Eff in $\mathcal{M}_{[\mathsf{Cause}]}$ if and only if it is a GPR cause in $\mathcal{N}$. By Lemma 4.2, Cause is a GPR cause for Eff in $\mathcal{M}_{[\mathsf{Cause}]}$ if and only if it is a GPR cause in $\mathcal{M}$. $\square$

Note, however, that the transformation changes the memory-requirements of schedulers witnessing that Cause is not a GPR cause for Eff. We will address the memory requirements in the original MDP later. With assumptions (A1)-(A3), condition (G) can be reformulated as follows:

Lemma 4.15. *Under assumptions (A1)-(A3),* Cause *satisfies (G) if and only if for each scheduler* $\mathfrak{S}$ *with* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Cause}) > 0$ *the following condition holds:*

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Cause}) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{unc}}) \; < \; \big(1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Cause})\big) \cdot \sum_{\mathsf{c}\in\mathsf{Cause}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{c}) \cdot w_{\mathsf{c}} \qquad \text{(GPR-1)}$$

With assumptions (A1)-(A3), a terminal state of $\mathcal{M}$ is reached almost surely under any scheduler after finitely many steps in expectation. Given a scheduler $\mathfrak{S}$ for $\mathcal{M}$ recall the definition of expected frequencies of state action-pairs (s, α), states $s \in S$ and state-sets $T \subseteq S$ under $\mathfrak{S}$:

$$freq_{\mathfrak{S}}(s, \alpha) \; \overset{\text{def}}{=} \; \mathrm{E}_{\mathcal{M}}^{\mathfrak{S}}(\text{number of visits to } s \text{ in which } \alpha \text{ is taken})$$

$$freq_{\mathfrak{S}}(s) \; \overset{\text{def}}{=} \; \sum_{\alpha\in Act(s)} freq_{\mathfrak{S}}(s, \alpha), \qquad freq_{\mathfrak{S}}(T) \overset{\text{def}}{=} \sum_{s\in T} freq_{\mathfrak{S}}(s).$$

Let T be one of the sets $\{\mathsf{eff}_{\mathsf{cov}}\}$, $\{\mathsf{eff}_{\mathsf{unc}}\}$, Cause, or a singleton $\{\mathsf{c}\}$ with $\mathsf{c} \in \mathsf{Cause}$. As T is visited at most once during each run of $\mathcal{M}$ (assumptions (A1) and (A2)), we have $\mathrm{Pr}_{\mathcal{N}}^{\mathfrak{S}}(\Diamond T) = freq_{\mathfrak{S}}(T)$ for each scheduler $\mathfrak{S}$. This allows us to express the violation of (G) in terms of a quadratic constraint system over variables for the expected frequencies of state-action pairs. Let *StAct* denote the set of state-action pairs in $\mathcal{M}$. We consider the following constraint system over the variables $x_{s,\alpha}$ for each $(s, \alpha) \in$ *StAct* where we use the short form notation $x_s = \sum_{\alpha\in Act(s)} x_{s,\alpha}$:

$$x_{s,\alpha} \; \geqslant \; 0 \qquad \text{for all } (s, \alpha) \in \textit{StAct} \tag{S1}$$

$$x_{\mathsf{init}} \; = \; 1 + \sum_{(t,\alpha)\in StAct} x_{t,\alpha} \cdot \mathrm{P}(t, \alpha, \mathsf{init}) \tag{S2}$$

$$x_s \; = \; \sum_{(t,\alpha)\in StAct} x_{t,\alpha} \cdot \mathrm{P}(t, \alpha, s) \qquad \text{for all } s \in S \setminus \{\mathsf{init}\} \tag{S3}$$

Using well-known results for MDPs without ECs (see, e.g., [Kal20, Theorem 9.16]), given a vector $x \in \mathbb{R}^{StAct}$, then x is a solution to (S1) and the balance equations (S2) and (S3) if and only if there is a (possibly history-dependent) scheduler $\mathfrak{S}$ for $\mathcal{M}$ with $x_{s,\alpha} = freq_{\mathfrak{S}}(s, \alpha)$ for all $(s, \alpha) \in$ *StAct* if and only if there is an MR-scheduler $\mathfrak{S}$ for $\mathcal{M}$ with $x_{s,\alpha} = freq_{\mathfrak{S}}(s, \alpha)$ for all $(s, \alpha) \in$ *StAct*.

The violation of (GPR-1) in Lemma 4.15 and the condition $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Cause}) > 0$ can be reformulated in terms of the frequency-variables as follows where x_{Cause} is an abbreviation for $\sum_{\mathsf{c}\in\mathsf{Cause}} x_{\mathsf{c}}$:

$$0 \; \geqslant \; \big(1 - x_{\mathsf{Cause}}\big) \cdot \sum_{\mathsf{c}\in\mathsf{Cause}} x_{\mathsf{c}} \cdot w_{\mathsf{c}} \; - \; x_{\mathsf{Cause}} \cdot x_{\mathsf{eff}_{\mathsf{unc}}} \tag{S4}$$

$$x_{\mathsf{Cause}} > 0 \tag{S5}$$

Lemma 4.16. *Under assumptions (A1)-(A3), the set* Cause *is not a GPR cause for* Eff *in* $\mathcal{M}$ *iff the constructed quadratic system of inequalities (S1)-(S5) has a solution.*

We can now prove that deciding the GPR condition can be done in coNP, Theorem 4.11.

Proof of Theorem 4.11. The quadratic system of inequalities can be constructed from $\mathcal{M}$, Cause, and Eff in polynomial time. Except for the strict inequality constraint in (S5), it has the form of a quadratic program, for which the threshold problem can be decided in NP by [Vav90]. We will prove that also with this strict inequality, it can be checked in NP whether the system (S1)-(S5) has a solution. As the system of inequalities is expressing the violation of (GPR), deciding whether a set Cause is a GPR cause can then be done in coNP.

To show that satisfiability of the system (S1)-(S5) is in NP, we will provide a non-deterministic algorithm that runs in polynomial time and finds a solution if one exists. Some of the arguments are similar to the arguments used in [Vav90]. Additionally, we will rely on the implicit function theorem.

We begin by proving what a solution to (S1)-(S5) can be assumed to look like. Thus assume that a solution to (S1)-(S5) exists. There are two possible cases:

Case 1: All solutions to (S1)-(S3) and (S5) satisfy (S4). Then, in particular, the frequency values of an MD-scheduler maximizing the probability to reach Cause are a solution to (S1)-(S3) and (S5) and hence to (S4) in this case.

Case 2: There are solutions to (S1)-(S3) and (S5) that violate (S4). The space of feasible points for conditions (S1)-(S3) and (S5) is connected. Furthermore, the right hand side of (S4)

$$\left(1 - x_{\mathsf{Cause}}\right) \cdot \sum_{c \in \mathsf{Cause}} x_c \cdot w_c \ - \ x_{\mathsf{Cause}} \cdot x_{\mathrm{eff}_{\mathrm{unc}}}$$

is continuous. Hence, as there are also solutions to (S1)-(S3) and (S5) that satisfy (S4) by assumption, there is a solution to (S1)-(S3) and (S5) that satisfies

$$\left(1 - x_{\mathsf{Cause}}\right) \cdot \sum_{c \in \mathsf{Cause}} x_c \cdot w_c \ - \ x_{\mathsf{Cause}} \cdot x_{\mathrm{eff}_{\mathrm{unc}}} = 0. \tag{S4'}$$

Now, let us take a closer look at Case 2: First of all, we add the equation

$$x_{\mathsf{Cause}} = \sum_{c \in \mathsf{Cause}} x_c \tag{S6}$$

to our system. Thus, the variables are x_{Cause} and $x_{s,\alpha}$ for each $(s, \alpha) \in \textit{StAct}$. Obviously, this does not influence the satisfiability. Equation (S4') now contains the new variable x_{Cause}, which is not an abbreviation anymore. We write x for the vector of variables $x_{s,\alpha}$ with $(s, \alpha) \in \textit{StAct}$.

In Case 2, there is a solution $(x^*, x^*_{\mathsf{Cause}})$ such that the maximal possible number of variables is 0 and such that x^*_{Cause} is maximal among all such solutions. Let X' be the set of variables that are 0 in $(x^*, x^*_{\mathsf{Cause}})$. We remove all variables from X' from all constraints by setting them to 0 and call the resulting system (T1)-(T6) where (T4) is obtained from (S4'), while all other equations (Ti) are obtained from (Si), by removing the chosen variables. We then collect the remaining variables in the vector $v = (y, y_{\mathsf{Cause}})$. Let $(y^*, y^*_{\mathsf{Cause}})$ be the solution $(x^*, x^*_{\mathsf{Cause}})$ after the variables in X' have been removed. Thus, all values in this vector are positive.

Define the function f as the right hand side of (T4):

$$f(y, y_{\mathsf{Cause}}) = \left(1 - y_{\mathsf{Cause}}\right) \cdot \sum_{c \in \mathsf{Cause}} y_c \cdot w_c \ - \ y_{\mathsf{Cause}} \cdot y_{\mathrm{eff}_{\mathrm{unc}}},$$

where the variables y are as the original variables x after the variables in X' have been removed. Now, we apply the implicit function theorem: Observe that

$$\frac{\partial f(y, y_{\text{Cause}})}{\partial y_{\text{Cause}}} = -\sum_{c \in \text{Cause}} y_c \cdot w_c - y_{\text{eff}_{\text{unc}}}.$$

Evaluated at $(y^*, y^*_{\text{Cause}})$, this value is non-zero as all summands are negative and there are at least some of the variables in the abbreviation y_c with $c \in \text{Cause}$ left, i.e., not removed because they were not 0 due to the original constraint (S5). So, we can apply the implicit function theorem, which guarantees us the existence of a function $g(y)$, such that $g(y^*) = y^*_{\text{Cause}}$ and, for all y' in an open ball B_1 around y^*, we have

$$f(y', g(y')) = 0.$$

By the implicit function theorem, we can explicitly compute the gradient

$$\nabla g = \left(\frac{\partial g(y)}{\partial y_1}, \ldots, \frac{\partial g(y)}{\partial y_k}\right) = -\left(\frac{\partial f(y, y_{\text{Cause}})}{\partial y_{\text{Cause}}}\right)^{-1} \cdot \left(\frac{\partial f(y, y_{\text{Cause}})}{\partial y_1}, \ldots, \frac{\partial f(y, y_{\text{Cause}})}{\partial y_k}\right)$$

of the derivatives on B_1 for the appropriate k from the derivatives of f. Note that on B_1, the gradient ∇g is 0 iff

$$H(y, y_{\text{Cause}}) \stackrel{\text{def}}{=} \left(\frac{\partial f(y, y_{\text{Cause}})}{\partial y_1}, \ldots, \frac{\partial f(y, y_{\text{Cause}})}{\partial y_k}\right)$$

is 0. Furthermore, all entries of $H(y, y_{\text{Cause}})$ are linear in the variables v as the function f is quadratic. As the function g has a local maximum in y^*, we know that ∇g evaluated at y^* is 0.

Equations (T2), (T3), and (T6) are linear equations in the remaining variables v. We can rewrite these three equations with a matrix M and a vector b whose entries can easily be expressed in terms of the coefficients of the original system (again, after the set of variables X' has been removed) as

$$Mv = b.$$

The solutions to this equation form an r-dimensional affine space W. It can be written as

$$W = \{\underbrace{c_0 + c_1 \cdot z_1 + \cdots + c_r \cdot z_r}_{\stackrel{\text{def}}{=} h(z_1, \ldots, z_r)} \mid (z_1, \ldots, z_r) \in \mathbb{R}^r\}$$

for some vectors $c_0, c_1, \ldots, c_r$ which can be computed from M and b in polynomial time.

Let B_2 be an open ball in $\mathbb{R}^r$ such that $h(B_2) \subseteq B_1$ and such that $h(B_2)$ contains $(y^*, y^*_{\text{Cause}})$. We claim that $g \circ h \colon B_2 \to \mathbb{R}$ has an isolated local maximum at $z^* \stackrel{\text{def}}{=} h^{-1}(y^*, y^*_{\text{Cause}})$. It is clear that $g \circ h$ has a local maximum since g has a local maximum at $(y^*, y^*_{\text{Cause}})$. Suppose now, that $g \circ h$ does not have an isolated local maximum at $h^{-1}(y^*, y^*_{\text{Cause}})$. As h is an affine map and the graph of g is the solution to a quadratic equation, this is only possible if there is a direction $d \in \mathbb{R}^r \setminus \{0\}$ such that

$$g \circ h(z^*) = g \circ h(z^* + t \cdot d) \tag{4.5}$$

for all $t \in \mathbb{R}$. Due to the boundedness of the polyhedron described by conditions (T1)-(T3), (T5) and (T6) and since z^* lies in the interior of this polyhedron, this means that there must be a value $q \in \mathbb{R}$ such that $(h(z^* + q \cdot d), g \circ h(z^* + q \cdot d))$ provides a solution v to equations (T1)-(T3), (T5) and (T6) with an additional 0. By the definition of g, this solution furthermore satisfies (T4) and, by equation (4.5), it still satisfies (T5). This contradicts the choice of the original solution $(x^*, x^*_{\text{Cause}})$.

So, $g \circ h \colon B_2 \to \mathbb{R}$ has an isolated local maximum at z^*. This implies that on an open ball around z^*, the point z^* is the only solution to

$$\nabla g(h(z)) = 0$$

and consequently, the only solution to

$$H(h(z)) = 0.$$

Since $H(h(z))$ is a vector of linear expressions in z, this implies that z^* is the only solution on $\mathbb{R}^r$ to $H(h(z)) = 0$. This is the key result that we need to provide a non-deterministic polynomial-time algorithm to check the satisfiability of the original constraint system.

Let us now describe the algorithm: The algorithm begins by computing the frequency values of an MD-scheduler as in Case 1 in polynomial time and checks whether the resulting vector of frequency values satisfies (S1)-(S5). If this is the case, the algorithm returns that the system is satisfiable.

If this is not the case, the algorithm tries to compute a solution to (S1)-(S3), (S5), and (S4') as in Case 2. The algorithm non-deterministically guesses a subset of the variables and removes them from all constraints by replacing them with 0.

Suppose we guess the set X' as above. We show that we then compute a solution. After the variables from X' have been removed, $H(y, y_{\mathsf{Cause}})$ can be computed in polynomial time as all the derivatives of f are linear expressions in the variables which require basic arithmetic and can be computed in polynomial time. Likewise, M and b can be computed in polynomial time from the original constraints after the guessed variables have been removed. The vectors $c_0, c_1, \ldots, c_r$ describing the solution space to $Mv = b$ can then also be computed in polynomial time.

Thus, also the vector $H(h(z))$ of linear expressions in the variables z can be computed in polynomial time. The equation system $H(h(z)) = 0$ has a unique solution if the guessed variables were indeed X'. In this case, the solution z^* can be computed in polynomial time as well. If the guess of variables was not X', then either there is no unique solution to this equation system which can be detected in polynomial time, or the solution, which is computed in the sequel in polynomial time, might not satisfy the original constraints, which is checked in the end.

From z^*, we can compute $y^* = h(z^*)$ using the vectors $c_0, c_1, \ldots, c_r$. The solution x^* is then obtained by plugging in 0s for the removed variables. Checking whether the resulting vector satisfies all constraints can also be done in polynomial time in the end. If X' was guessed correctly, this vector x^* indeed forms a solution to the original constraints as we have seen.

In summary, the algorithm needs to guess the set X' of variables which are 0 in a solution to the original constraints with the maximal number of zeroes. All other steps are deterministic polynomial-time computations. Thus, satisfiability of (S1)-(S5) can be checked in NP. □

Memory requirements of schedulers in the original MDP (Proof of Theorem 4.12). Every solution to the linear system of inequalities (S1), (S2), and (S3) corresponds to the expected frequencies of state-action pairs of an MR-scheduler in the transformed model satisfying (A1)-(A3). Hence:

Corollary 4.17. *Under assumptions (A1)-(A3),* Cause *is no GPR cause for* Eff *iff there exists an MR-scheduler $\mathfrak{T}$ with* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond \mathsf{Cause}) > 0$ *violating* (GPR).

The model transformation we used for assumptions (A1)-(A3), however, does affect the memory requirements of scheduler. We may further restrict the MR-schedulers necessary to witness non-causality under assumptions (A1)-(A3). For the following lemma, recall that τ is the action of the MEC quotient used for the extra transition from states representing MECs to a new trap state (see also assumption (A3)).

Lemma 4.18. *Assume (A1)-(A3). Given an MR-scheduler* $\mathfrak{U}$ *with* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge\mathsf{Cause}) > 0$ *that violates* (GPR), *an MR-scheduler* $\mathfrak{T}$ *with* $\mathfrak{T}(s)(\tau) \in \{0,1\}$ *for each state* s *with* $\tau \in Act(s)$ *that satifies* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\lozenge\mathsf{Cause}) > 0$ *and violates* (GPR) *is computable in polynomial time.*

Proof. Let $\mathfrak{U}$ be a scheduler with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge\mathsf{Cause}) > 0$ violating (GPR-1), i.e.:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge\mathsf{Cause}) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge\mathsf{eff}_{\mathsf{unc}}) \; < \; \big(1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge\mathsf{Cause})\big) \cdot \sum_{c \in \mathsf{Cause}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\lozenge c) \cdot w_c.$$

We will show how to transform $\mathfrak{U}$ into an MR-scheduler $\mathfrak{T}$ that schedules the τ-transitions to $\mathsf{noeff}_{\mathsf{tn}}$ with probability 0 or 1. We regard the set U of states u which have a τ-transition to $\mathsf{noeff}_{\mathsf{tn}}$ (recall that then $P(u, \tau, \mathsf{noeff}_{\mathsf{tn}}) = 1$) and where $0 < \mathfrak{U}(u)(\tau) < 1$. We now process the U-states in an arbitrary order, say $u_1, \ldots, u_k$, and generate a sequence $\mathfrak{T}_0 = \mathfrak{U}, \mathfrak{T}_1, \ldots, \mathfrak{T}_k$ of MR-schedulers such that for $i \in \{1, \ldots, k\}$:

- $\mathfrak{T}_i$ refutes (GPR) (or equivalently condition (GPR-1) from Lemma 4.15)
- $\mathfrak{T}_i$ agrees with $\mathfrak{T}_{i-1}$ for all states but u_i,
- $\mathfrak{T}_i(u_i)(\tau) \in \{0,1\}$.

Thus, the final scheduler $\mathfrak{T}_k$ satisfies the desired properties.

To explain how to derive $\mathfrak{T}_i$ from $\mathfrak{T}_{i-1}$, let $i \in \{1, \ldots, k\}$, $\mathfrak{V} = \mathfrak{T}_{i-1}$, $u = u_i$ and $y = 1 - \mathfrak{V}(u)(\tau)$. Then, $0 < y < 1$ (as $u \in U$ and by definition of U) and $y = \sum_{\alpha \in Act(u) \setminus \{\tau\}} \mathfrak{V}(u)(\alpha)$. For $x \in [0,1]$, let $\mathfrak{V}_x$ denote the MR-scheduler that agrees with $\mathfrak{V}$ for all states but u, for which $\mathfrak{V}_x$'s decision is:

$$\mathfrak{V}_x(u)(\tau) = 1 - x, \qquad \mathfrak{V}_x(u)(\alpha) = \mathfrak{V}(u)(\alpha) \cdot \frac{x}{y} \quad \text{for } \alpha \in Act(u) \setminus \{\tau\}$$

Obviously, $\mathfrak{V}_y = \mathfrak{V}$. We now show that at least one of the two MR-schedulers $\mathfrak{V}_0$ or $\mathfrak{V}_1$ also refutes (GPR). For this, we suppose by contraction that this is not the case, which means that (GPR) holds for both. Let $f : [0,1] \to \mathbb{Q}$ be defined by

$$f(x) \; = \; \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge\mathsf{Cause}) \cdot \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge\mathsf{eff}_{\mathsf{unc}}) - \big(1 - \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge\mathsf{Cause})\big) \cdot \sum_{c \in \mathsf{Cause}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge c) \cdot w_c$$

As $\mathfrak{V} = \mathfrak{V}_y$ violates (GPR-1), while $\mathfrak{V}_0$ and $\mathfrak{V}_1$ satisfy (GPR-1) we obtain:

$$f(0), f(1) < 0 \qquad \text{and} \qquad f(y) \geqslant 0$$

We now split Cause into the set C of states $c \in \mathsf{Cause}$ such that there is a $\mathfrak{V}$-path from init to c that traverses u and $D = \mathsf{Cause} \setminus C$. Thus, $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge\mathsf{Cause}) = p_x + p$ where $p_x = \mathrm{Pr}^{\mathfrak{V}_x}(\lozenge C)$ and $p = \mathrm{Pr}^{\mathfrak{V}}(\lozenge D)$. Similarly, $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge\mathsf{eff}_{\mathsf{unc}})$ has the form $q_x + q$ where $q_x = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_x}(\lozenge(u \wedge \lozenge\mathsf{eff}_{\mathsf{unc}}))$ and $q = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}_x}((\neg u)\,U\,\mathsf{eff}_{\mathsf{unc}})$. With $p_{x,c} = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}_x}(\lozenge c)$ for $c \in C$ and $p_d = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{V}}(\lozenge d)$ for $d \in D$, let

$$v_x = \sum_{c \in C} p_{x,c} \cdot w_c \qquad \text{and} \qquad v = \sum_{d \in D} p_d \cdot w_d$$

As y is fixed, the values $p_y, p_{y,c}, q_y, v_y$ can be seen as constants. Moreover, $p_x, p_{x,c}, q_x, v_x$ differ from $p_y, p_{y,c}, q_y, v_y$ only by the factor $\frac{x}{y}$. That is:

$$p_x = p_y \frac{x}{y}, \quad p_{x,c} = p_{y,c} \frac{x}{y}, \quad q_x = q_y \frac{x}{y} \quad \text{and} \quad v_x = v_y \frac{x}{y}.$$

Thus, $f(x)$ has the following form:

$$
\begin{aligned}
f(x) &= (p_x+p)(q_x+q) - \big(1-(p_x+p)\big)(\nu_x+\nu) \\
&= \underbrace{p_x q_x + p_x \nu_x}_{\mathfrak{a}x^2} + \underbrace{p_x(q+\nu) + q_x p - \nu_x}_{\mathfrak{b}x} + \underbrace{pq - \nu + p\nu}_{\mathfrak{c}} \\
&= \mathfrak{a}x^2 + \mathfrak{b}x + \mathfrak{c}
\end{aligned}
$$

For the value $\mathfrak{a}$, we have $\mathfrak{a}x^2 = p_x q_x + p_x \nu_x$ and hence $\mathfrak{a} = \frac{1}{y^2}(p_y q_y + p_y \nu_y) > 0$. But then the second derivative $f''(x) = 2\mathfrak{a}$ of f is positive, which yields that f has a global minimum at some point x_0 and is strictly decreasing for $x < x_0$ and strictly increasing for $x > x_0$. As $f(0)$ and $f(1)$ are both negative, we obtain $f(x) < 0$ for all x in the interval $[0,1]$. But this contradicts $f(y) \geqslant 0$.

This yields that at least one of the schedulers $\mathfrak{V}_0$ or $\mathfrak{V}_1$ witnesses the violation of (GPR). Thus, we can define $\mathfrak{T}_i \in \{\mathfrak{V}_0, \mathfrak{V}_1\}$ accordingly.

The number of states k in U is bounded by the number of states in S. In each iteration of the above construction, the function value $f(0)$ is sufficient to determine one of the schedulers $\mathfrak{V}_0$ and $\mathfrak{V}_1$ witnessing the violation of (GPR). So, the procedure has to compute the values in condition (GPR-1) for k-many MR-schedulers and update the scheduler afterwards. As the update can easily be carried out in polynomial time, the run-time of all k iterations is polynomial as well. $\qquad\square$

The condition that τ only has to be scheduled with probability 0 or 1 in each state is the key to transfer the sufficiency of MR-schedulers to the MDP $\mathcal{M}_{[\mathsf{Cause}]}$. This fact is of general interest as well and stated in the following theorem where τ again is the action added to move from a state $s_{\mathcal{E}}$ to the new trap state in the MEC-quotient.

Theorem 4.19. *Let $\mathcal{M}$ be an MDP with pairwise disjoint action sets for all states. Then, for each MR-scheduler $\mathfrak{S}$ for the MEC-quotient of $\mathcal{M}$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) \in \{0,1\}$ for each MEC $\mathcal{E}$ of $\mathcal{M}$ there is an MR-scheduler $\mathfrak{T}$ for $\mathcal{M}$ such that every action α of $\mathcal{M}$ that does not belong to an MEC of $\mathcal{M}$, has the same expected frequency under $\mathfrak{S}$ and $\mathfrak{T}$.*

Proof. Let $\mathfrak{S}$ be an MR-scheduler for $MEC(\mathcal{M})$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) \in \{0,1\}$ for each MEC $\mathcal{E}$ of $\mathcal{M}$. We consider the following extension $\mathcal{M}'$ of $\mathcal{M}$: The state space of $\mathcal{M}$ is extended by a new terminal state $\perp$ and a fresh action τ is enabled in each state s that belongs to a MEC of $\mathcal{M}$. Action τ leads to $\perp$ with probability 1. All remaining transition probabilities are as in $\mathcal{M}$. So, $\mathcal{M}'$ is obtained from $\mathcal{M}$ by allowing a transition to a new terminal state $\perp$ from each state that belongs to a MEC.

Now, we first provide a finite-memory scheduler $\mathfrak{T}$ for $\mathcal{M}'$ that leaves each MEC $\mathcal{E}$ for which $\mathfrak{S}(s_{\mathcal{E}})(\tau) = 0$ via the state action pair (s, α) with probability $\mathfrak{S}(s_{\mathcal{E}})(\alpha)$. Recall that we assume that each action is enabled in at most one state and that the actions enabled in the state $s_{\mathcal{E}}$ in $MEC(\mathcal{M})$ are precisely the actions that are enabled in some state of $\mathcal{E}$ and that do not belong to $\mathcal{E}$ (Section 2.3)

Let us define the scheduler $\mathfrak{T}$: In all states that do not belong to a MEC $\mathcal{E}$ of $\mathcal{M}$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) = 0$, the behavior of $\mathfrak{T}$ is memoryless: For each state s of $\mathcal{M}$ (and hence of $\mathcal{M}'$) that does not belong to a MEC, $\mathfrak{T}(s) = \mathfrak{S}(s)$. For each state s in an end component $\mathcal{E}$ of $\mathcal{M}$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) = 1$, we define $\mathfrak{T}(s)(\tau) = 1$. If a MEC $\mathcal{E}$ of $\mathcal{M}$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) = 0$ is entered, $\mathfrak{T}$ makes use of finitely many memory modes as follows: Enumerate the state action pairs (s, α) where s belongs to $\mathcal{E}$, but α does not belong to $\mathcal{E}$, and for which $\mathfrak{S}(s_{\mathcal{E}})(\alpha) > 0$ by $(s_1, \alpha_1), \ldots, (s_k, \alpha_k)$. Further, let $p_i \overset{\text{def}}{=} \mathfrak{S}(s_{\mathcal{E}})(\alpha_i) > 0$ for all $1 \leqslant i \leqslant k$. By assumption $\sum_{1 \leqslant i \leqslant k} p_i = 1$. When entering $\mathcal{E}$, the scheduler works in k memory modes $1, \ldots, k$ until an action α that does not belong to $\mathcal{E}$ is scheduled starting in memory mode 1. In each memory mode i, $\mathfrak{T}$ follows an MD-scheduler for $\mathcal{E}$ that reaches s_i with probability 1 from all

states of $\mathcal{E}$. Once, s_i is reached, $\mathfrak{T}$ chooses action α_i with probability

$$q_i \overset{\text{def}}{=} \frac{p_i}{1 - \sum_{j < i} p_j}.$$

Now $\mathfrak{T}$ leaves $\mathcal{E}$ via (s_k, α_k) with probability 1 if it reaches the last memory mode k. As $\mathfrak{T}$ behaves MD in each mode, it leaves the end component $\mathcal{E}$ after finitely many steps in expectation. Furthermore, for each $i \leqslant k$, it leaves $\mathcal{E}$ via (s_i, α_i) with probability $(1 - \sum_{j < i} p_j) \cdot q_i = p_i$. Since the behavior of $\mathfrak{S}$ in $MEC(\mathcal{M})$ is mimicked by $\mathfrak{T}$ in $\mathcal{M}'$, we conclude that the expected frequency of actions in $\mathcal{M}$ which do not belong to an end component is the same in $\mathcal{M}'$ under $\mathfrak{T}$ and in $MEC(\mathcal{M})$ under $\mathfrak{S}$.

The expected frequency of each state-action pair of $\mathcal{M}'$ under $\mathfrak{T}$ is finite, since each MEC of $\mathcal{M}'$ is left after finitely many steps in expectation. In the terminology of [Kal20], the scheduler $\mathfrak{T}$ is *transient*. By [Kal20, Theorem 9.16], this implies that there is a MR-scheduler $\mathfrak{U}$ for $\mathcal{M}'$ under which the expected frequency of state-action pairs is the same as under $\mathfrak{T}$ and thus the expected frequency in $\mathcal{M}'$ of actions α of $\mathcal{M}$ that do not belong to an end component is the same as under $\mathfrak{S}$ in $MEC(\mathcal{M})$.

Finally, we modify $\mathfrak{U}$ such that it becomes a scheduler for $\mathcal{M}$: For each end component $\mathcal{E}$ of $\mathcal{M}$ with $\mathfrak{S}(s_{\mathcal{E}})(\tau) = 1$, we fix a memoryless scheduler $\mathfrak{U}_{\mathcal{E}}$ that does not leave the end component. Now, whenever a state s in such an end component is visited, the modified scheduler switches to the behavior of $\mathfrak{U}_{\mathcal{E}}$ instead of choosing action τ with probability 1. Clearly, this does not affect the expected frequency of actions of $\mathcal{M}$ that do not belong to an end component and hence the modified scheduler is as claimed in the theorem. $\qquad\square$

Remark 4.20. The proof of Theorem 4.19 above provides an algorithm how to obtain the scheduler $\mathfrak{T}$ from $\mathfrak{S}$. The number of memory modes of the intermediately constructed finite-memory scheduler is bounded by the number of state-action pairs of $\mathcal{M}$. Further, in each memory mode during the traversal of a MEC, the scheduler behaves in a memoryless deterministic way. Hence, the induced Markov chain is of size polynomial in the size of the MDP $\mathcal{M}$ and the representation of the scheduler $\mathfrak{S}$. Therefore, also the expected frequencies of all state-action pairs under the intermediate finite-memory scheduler and hence under $\mathfrak{T}$ can be computed in time polynomial in the size of the MDP $\mathcal{M}$ and the representation of the scheduler $\mathfrak{S}$. So, also the scheduler $\mathfrak{T}$ itself which can be derived from these expected frequencies can be computed in polynomial time from $\mathfrak{S}$.

Together with Lemma 4.18, this means that $\mathfrak{T}$ and hence the scheduler with two memory modes whose existence is stated in Theorem 4.12 can be computed from a solution to the constraint system (S1)-(S5) from Section 4.2 in time polynomial in the size of the original MDP and the size of the representation of the solution to (S1)-(S5). $\qquad\triangleleft$

With these results we can now prove the second main result of this section, Theorem 4.12, stating that if (GPR) does not hold there is a finite-memory scheduler with two memory cells refuting the GPR condition.

Proof of Theorem 4.12. The model transformation establishing assumptions (A1)-(A3) results in the MEC-quotient of $\mathcal{M}_{[\mathsf{Cause}]}$ up to the renaming and collapsing of terminal states. By Corollary 4.17 and Theorem 4.19, we conclude that Cause is not a GPR cause for Eff in $\mathcal{M}$ if and only if there is a MR-scheduler $\mathfrak{S}$ for $\mathcal{M}_{[\mathsf{Cause}]}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\Diamond\mathsf{Cause}) > 0$ that violates (GPR). As in Remark 4.9, $\mathfrak{S}$ can be extended to a finite-memory randomized scheduler $\mathfrak{T}$ for $\mathcal{M}$ with two memory cells. $\qquad\square$

Remark 4.21 (On lower bounds on GPR checking). Solving systems of quadratic inequalities with linear side constraints is NP-hard in general (see, e.g., [GJ79]). For convex problems, in which the associated symmetric matrix occurring in the quadratic inequality has only non-negative eigenvalues, the problem is, however, solvable in polynomial time [KTK80]. Unfortunately, the

quadratic constraint system describing a scheduler refuting (GPR) given by (S1)-(S5) is not of this form. We observe that even if Cause is a singleton $\{c\}$ and the variable $x_{\text{eff}_{\text{unc}}}$ is forced to take a constant value y by (S1)-(S3), i.e., by the structure of the MDP, the inequality (S4) takes the form:

$$x_c \cdot w_c - x_c^2 \cdot (w_c + y) \leqslant 0 \qquad (4.6)$$

Here, the 1×1-matrix $(-w_c - y)$ has a negative eigenvalue. Although it is not ruled out that (S1)-(S5) belongs to another class of efficiently solvable constraint systems, the NP-hardness result in [PV91] for the solvability of quadratic inequalities of the form (4.6) with linear side constraints might be an indication for the computational difficulty.   ◁

5. QUALITY AND OPTIMALITY OF CAUSES

The goal of this section is to identify notions that measure how "good" causes are and to present algorithms to determine good causes according to the proposed quality measures. We have seen so far that small (singleton) causes are easy to determine (see Section 4.1). Moreover, it is easy to see that the proposed existence-checking algorithm can be formulated in such a way that the algorithm returns a singleton (strict or global) probability-raising cause $\{c_0\}$ with maximal *precision*, i.e., a state c_0 where $\inf_{\mathfrak{S}} \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Eff}|\Diamond c_0) = \Pr_{\mathcal{M},c_0}^{\min}(\Diamond \text{Eff})$ is maximal. On the other hand, singleton or small cause sets might have poor coverage in the sense that the probability for paths that reach an effect state without visiting a cause state before ("uncovered effects") can be large. This motivates the consideration of quality notions for causes that incorporate how well effect scenarios are covered. We take inspiration of quality measures that are considered in statistical analysis (see e.g. [Pow11]). This includes the *recall* as a measure for the relative coverage (proportion of covered effects among all effect scenarios), the *coverage ratio* (quotient of covered and uncovered effects) as well as the *f-score*. The f-score is a standard measure for classifiers defined by the harmonic mean of precision and recall. It can be seen as a compromise to achieve both good precision and good recall.

In this section, we assume as before an MDP $\mathcal{M} = (S, Act, P, \text{init})$ and $\text{Eff} \subseteq S$ are given where all effect states are terminal. Furthermore, we suppose all states $s \in S$ are reachable from init.

5.1. **Quality measures for causes.** In statistical analysis, the precision of a classifier with binary outcomes ("positive" or "negative") is defined as the ratio of all true positives among all positively classified elements, while its recall is defined as the ratio of all true positives among all actual positive elements. Translated to our setting, we consider classifiers induced by a given cause set Cause that return "positive" for sample paths in case that a cause state is visited and "negative" otherwise. The intuitive meaning of true positives, false positives, true negatives and false negatives is as described in the confusion matrix in Figure 13. The formal definition is

$$\begin{aligned} \text{tp}^{\mathfrak{S}} &= \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Cause} \wedge \Diamond \text{Eff}), & \text{tn}^{\mathfrak{S}} &= \Pr_{\mathcal{M}}^{\mathfrak{S}}(\neg\Diamond \text{Cause} \wedge \neg\Diamond \text{Eff}), \\ \text{fp}^{\mathfrak{S}} &= \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Cause} \wedge \neg\Diamond \text{Eff}), & \text{fn}^{\mathfrak{S}} &= \Pr_{\mathcal{M}}^{\mathfrak{S}}(\neg\Diamond \text{Cause} \wedge \Diamond \text{Eff}). \end{aligned}$$

Path hits	Eff	¬Eff
Cause	True positive (tp)	False positive (fp)
	Cause correctly predicted Eff	Cause falsely predicted Eff
¬Cause	False negative (fn)	True negative (tn)
	Cause falsely not predicted Eff	Cause correctly not predicted Eff

Figure 13: Confusion matrix for Cause as a binary classifier for Eff

With this interpretation of causes as binary classifiers in mind, the recall and precision and coverage ratio of a cause set Cause *under a scheduler* $\mathfrak{S}$ are defined as follows:

$$
\begin{aligned}
precision^{\mathfrak{S}}(\mathsf{Cause}) &= \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\, \lozenge\mathsf{Eff} \mid \lozenge\mathsf{Cause}\,) &&= \frac{\mathsf{tp}^{\mathfrak{S}}}{\mathsf{tp}^{\mathfrak{S}} + \mathsf{fp}^{\mathfrak{S}}} \\
recall^{\mathfrak{S}}(\mathsf{Cause}) &= \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\, \lozenge\mathsf{Cause} \mid \lozenge\mathsf{Eff}\,) &&= \frac{\mathsf{tp}^{\mathfrak{S}}}{\mathsf{tp}^{\mathfrak{S}} + \mathsf{fn}^{\mathfrak{S}}} \\
covrat^{\mathfrak{S}}(\mathsf{Cause}) &= \frac{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}\big(\lozenge(\mathsf{Cause} \wedge \lozenge\mathsf{Eff})\big)}{\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}\big((\neg\lozenge\mathsf{Cause}) \wedge \lozenge\mathsf{Eff}\big)} &&= \frac{\mathsf{tp}^{\mathfrak{S}}}{\mathsf{fn}^{\mathfrak{S}}}.
\end{aligned}
$$

Note that for these definitions we make some respective assumptions on the scheduler. We assume
- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Cause}) > 0$ for $precision^{\mathfrak{S}}(\mathsf{Cause})$,
- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) > 0$ for $recall^{\mathfrak{S}}(\mathsf{Cause})$ and
- $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}\big((\neg\lozenge\mathsf{Cause}) \wedge \lozenge\mathsf{Eff}\big) > 0)$ for $covrat^{\mathfrak{S}}(\mathsf{Cause})$.

If we have $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}\big((\neg\lozenge\mathsf{Cause}) \wedge \lozenge\mathsf{Eff}\big) = 0$ and $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Cause}) > 0$ for some scheduler $\mathfrak{S}$, we define $covrat^{\mathfrak{S}}(\mathsf{Cause}) = +\infty$. This makes sense since we can converge to such a scheduler $\mathfrak{S}$ with a sequence of schedulers $\mathfrak{S}_0 \ldots$ for which $\mathrm{Pr}^{\mathfrak{S}_{\mathfrak{i}}}_{\mathcal{M}}\big((\neg\lozenge\mathsf{Cause}) \wedge \lozenge\mathsf{Eff}\big) > 0$ and $\mathrm{Pr}^{\mathfrak{S}_{\mathfrak{i}}}_{\mathcal{M}}(\lozenge\mathsf{Cause}) > 0$ for $\mathfrak{i} \in \mathbb{N}$. The coverage ratio of such a sequence converges to $+\infty$.

Finally, the f-score of Cause *under a scheduler* $\mathfrak{S}$ is defined as the harmonic mean of the precision and recall. Here we assume $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Cause}) > 0$, which implies $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) > 0$:

$$
fscore^{\mathfrak{S}}(\mathsf{Cause}) \stackrel{\mathrm{def}}{=} 2 \cdot \frac{precision^{\mathfrak{S}}(\mathsf{Cause}) \cdot recall^{\mathfrak{S}}(\mathsf{Cause})}{precision^{\mathfrak{S}}(\mathsf{Cause}) + recall^{\mathfrak{S}}(\mathsf{Cause})} = \frac{2 \cdot \mathsf{tp}^{\mathfrak{S}}}{2 \cdot \mathsf{tp}^{\mathfrak{S}} + \mathsf{fp}^{\mathfrak{S}} + \mathsf{fn}^{\mathfrak{S}}}
$$

If, however, $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) > 0$ and $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Cause}) = 0$ for some $\mathfrak{S}$, define $fscore^{\mathfrak{S}}(\mathsf{Cause}) = 0$. This again makes sense as for a sequence of schedulers converging to $\mathfrak{S}$ the f-score also converges to 0 (also see Lemma 5.6).

To lift the definitions of the quality measures under a scheduler to the quality measure of a cause, we consider the worst-case scheduler:

Definition 5.1 (Quality measures for causes). Let Cause be a PR cause. We define
$$
recall(\mathsf{Cause}) = \inf_{\mathfrak{S}} recall^{\mathfrak{S}}(\mathsf{Cause}) = \mathrm{Pr}^{\min}_{\mathcal{M}}(\, \lozenge\mathsf{Cause} \mid \lozenge\mathsf{Eff}\,)
$$
when ranging over all schedulers $\mathfrak{S}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\lozenge\mathsf{Eff}) > 0$. Likewise, the coverage ratio and f-score of Cause are defined by the worst-case coverage ratio resp. f-score – ranging over schedulers for which $covrat^{\mathfrak{S}}(\mathsf{Cause})$ resp. $fscore^{\mathfrak{S}}(\mathsf{Cause})$ is defined:
$$
covrat(\mathsf{Cause}) = \inf_{\mathfrak{S}} covrat^{\mathfrak{S}}(\mathsf{Cause}), \quad fscore(\mathsf{Cause}) = \inf_{\mathfrak{S}} fscore^{\mathfrak{S}}(\mathsf{Cause}) \qquad \lhd
$$

Besides the quality measures defined so far, which we will address in detail, there is a vast landscape of further quality measures for binary classifiers in the literature (for an overview, see, e.g., [Pow11]). One prominent example which has been claimed to be superior to the f-score recently

[CJ20] is *Matthews correlation coefficient* (MCC). In terms of the entries of a confusion matrix (as in Figure 13), it is defined as

$$\mathrm{MCC} = \frac{\mathrm{tp} \cdot \mathrm{tn} - \mathrm{fp} \cdot \mathrm{fn}}{\sqrt{(\mathrm{tp} + \mathrm{fp}) \cdot (\mathrm{tp} + \mathrm{fn}) \cdot (\mathrm{tn} + \mathrm{fp}) \cdot (\mathrm{tn} + \mathrm{fn})}}$$

In contrast to the f-score (as well as recall and coverage ratio), it makes use of all four entries of the confusion matrix. In our setting, we could assign the MCC to a Cause by again taking the infimum of the value over all sensible schedulers.

Like the MCC, almost all (cf. [Pow11]) of the quality measures studied in the literature are algebraic functions (intuitively speaking, built from polynomials, fractions and root functions) in the entries of the confusion matrix. At the end of this section, we will comment on the computational properties of finding good causes when quality is measured by the infimum over all sensible schedulers of an algebraic function in the entries of the confusion matrix.

5.2. **Computation schemes for the quality measures for fixed cause set.** For this section, we assume a fixed PR cause Cause is given and address the problem to compute its quality values. The first observation is, that all quality measures are preserved by the switch from $\mathcal{M}$ to $\mathcal{M}_{[\mathsf{Cause}]}$ as well as the transformations of $\mathcal{M}_{[\mathsf{Cause}]}$ to an MDP that satisfies conditions (A1)-(A3) of Section 4.2. In the following Lemmata 5.2 and 5.3 we show that the quality measures *recall*, *covrat* and *fscore* of a fixed Cause are compatible with the model transformations from section 4. These are, on one hand a transformation to $\mathcal{M}_{[\mathsf{Cause}]}$, which only considers the minimal probability to reach Eff starting from Cause, and on the other hand a transformation to an MDP $\mathcal{N}$ satisfying (A1)-(A3), which has no end components and has exactly four terminal states $\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}, \mathsf{noeff}_{\mathsf{fp}}, \mathsf{noeff}_{\mathsf{tn}}$.

Lemma 5.2. *If* Cause *is an SPR or a GPR cause then:*

$$
\begin{aligned}
\mathit{recall}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{recall}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) \\
\mathit{covrat}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{covrat}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) \\
\mathit{fscore}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{fscore}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})
\end{aligned}
$$

Proof. "$\leqslant$": A scheduler for $\mathcal{M}_{[\mathsf{Cause}]}$ can be seen as a scheduler $\mathfrak{S}$ for $\mathcal{M}$ behaving as an MD-scheduler minimizing the reachability probability of Eff from every state in Cause and we have:

$$
\begin{aligned}
\mathit{recall}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{recall}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) \\
\mathit{covrat}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{covrat}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) \\
\mathit{precision}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) &= \mathit{precision}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})
\end{aligned}
$$

and therefore:

$$\mathit{fscore}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) = \mathit{fscore}^{\mathfrak{S}}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})$$

We obtain $\mathit{recall}_{\mathcal{M}}(\mathsf{Cause}) \leqslant \mathit{recall}_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})$ and the analogous statements for the coverage ratio and the f-score.

"$\geqslant$": Let $\mathfrak{S}$ be a scheduler of $\mathcal{M}$. Let $\mathfrak{T}$ be the scheduler of $\mathcal{M}$ that behaves as $\mathfrak{S}$ until the first visit to a state in Cause. As soon as $\mathfrak{T}$ has reached Cause, it behaves as an MD-scheduler minimizing the probability to reach Eff. Recall and coverage under $\mathfrak{T}$ and $\mathfrak{S}$ have the form:

$$
\begin{array}{llll}
\mathit{recall}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) &= \dfrac{x}{x+q} & \quad \mathit{covrat}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{Cause}) &= \dfrac{x}{q} \\[2ex]
\mathit{recall}^{\mathfrak{T}}_{\mathcal{M}}(\mathsf{Cause}) &= \dfrac{y}{y+q} & \quad \mathit{covrat}^{\mathfrak{T}}_{\mathcal{M}}(\mathsf{Cause}) &= \dfrac{y}{q}
\end{array}
$$

where $x \geqslant y$ (and $q = \mathrm{fn}^{\mathfrak{S}}$). Considering $\mathfrak{T}$ as a scheduler of $\mathcal{M}$ and of $\mathcal{M}_{[\mathsf{Cause}]}$, we get:

$$
\begin{aligned}
recall_{\mathcal{M}}^{\mathfrak{S}}(\mathsf{Cause}) &\geqslant & recall_{\mathcal{M}}^{\mathfrak{T}}(\mathsf{Cause}) &= & recall_{\mathcal{M}_{[\mathsf{Cause}]}}^{\mathfrak{T}}(\mathsf{Cause}) \\
covrat_{\mathcal{M}}^{\mathfrak{S}}(\mathsf{Cause}) &\geqslant & covrat_{\mathcal{M}}^{\mathfrak{T}}(\mathsf{Cause}) &= & covrat_{\mathcal{M}_{[\mathsf{Cause}]}}^{\mathfrak{T}}(\mathsf{Cause})
\end{aligned}
$$

This implies:

$$
\begin{aligned}
recall_{\mathcal{M}}^{\mathfrak{S}}(\mathsf{Cause}) &\geqslant & recall_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) \\
covrat_{\mathcal{M}}(\mathsf{Cause}) &\geqslant & covrat_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})
\end{aligned}
$$

With similar arguments we get:

$$
precision_{\mathcal{M}}^{\mathfrak{S}}(\mathsf{Cause}) \quad \geqslant \quad precision_{\mathcal{M}}^{\mathfrak{T}}(\mathsf{Cause}) \quad = \quad precision_{\mathcal{M}_{[\mathsf{Cause}]}}^{\mathfrak{T}}(\mathsf{Cause})
$$

As the harmonic mean viewed as a function $f : \mathbb{R}_{>0}^2 \to \mathbb{R}$, $f(x, y) = 2\frac{xy}{x+y}$ is monotonically increasing in both arguments (note that $\frac{\mathrm{d}f}{\mathrm{d}x} = \frac{y^2}{(x+y)^2} > 0$ and $\frac{\mathrm{d}f}{\mathrm{d}y} = \frac{x^2}{(x+y)^2} > 0$), we obtain:

$$
fscore_{\mathcal{M}}^{\mathfrak{S}}(\mathsf{Cause}) \;\geqslant\; fscore_{\mathcal{M}}^{\mathfrak{T}}(\mathsf{Cause}) \;=\; fscore_{\mathcal{M}_{[\mathsf{Cause}]}}^{\mathfrak{T}}(\mathsf{Cause})
$$

This yields $fscore_{\mathcal{M}}(\mathsf{Cause}) \geqslant fscore_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause})$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Lemma 5.3. *Let $\mathcal{N}$ be the MEC-quotient of $\mathcal{M}_{[\mathsf{Cause}]}$ for some MDP $\mathcal{M}$ with a set of terminal states* Eff *and an SPR or a GPR cause* Cause. *Then:*

$$
\begin{aligned}
recall_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) &= & recall_{\mathcal{N}}(\mathsf{Cause}) \\
covrat_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) &= & covrat_{\mathcal{N}}(\mathsf{Cause}) \\
fscore_{\mathcal{M}_{[\mathsf{Cause}]}}(\mathsf{Cause}) &= & fscore_{\mathcal{N}}(\mathsf{Cause})
\end{aligned}
$$

Proof. Analogously to the proof of Lemma 4.13. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

This now allows us to work under assumptions (A1)-(A3) when addressing problems concerning the quality measures for a fixed cause set.

As efficient computation methods for $recall(\mathsf{Cause})$ are known from literature (see [BKKM14, Mär20] for poly-time algorithms to compute conditional reachability probabilities), we can use the same methods to compute the coverage ratio.

Corollary 5.4. *The value* covrat(Cause) *and corresponding worst-case schedulers are computable in polynomial time.*

Proof. For a given scheduler $\mathfrak{S}$ we have

$$
covrat^{\mathfrak{S}}(\mathsf{Cause}) = \frac{\mathrm{tp}^{\mathfrak{S}}}{\mathrm{fn}^{\mathfrak{S}}} \quad \text{and} \quad recall^{\mathfrak{S}}(\mathsf{Cause}) = \frac{\mathrm{tp}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}} + \mathrm{fn}^{\mathfrak{S}}}
$$

We thus get the following

$$
\frac{1}{\frac{\mathrm{tp}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}} + \mathrm{fn}^{\mathfrak{S}}}} = \frac{\mathrm{tp}^{\mathfrak{S}} + \mathrm{fn}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}}} = \frac{\mathrm{tp}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}}} + \frac{\mathrm{fn}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}}} = 1 + \frac{\mathrm{fn}^{\mathfrak{S}}}{\mathrm{tp}^{\mathfrak{S}}} = 1 + \frac{1}{\frac{\mathrm{tp}^{\mathfrak{S}}}{\mathrm{fn}^{\mathfrak{S}}}}.
$$

This implies

$$
\frac{1}{recall^{\mathfrak{S}}(\mathsf{Cause})} = 1 + \frac{1}{covrat^{\mathfrak{S}}(\mathsf{Cause})} \quad \text{and thus} \quad covrat^{\mathfrak{S}}(\mathsf{Cause}) = 1 / \left(\frac{1}{recall^{\mathfrak{S}}(\mathsf{Cause})} - 1 \right)
$$

Computing *covrat*(Cause) now implores us to take the infimum of all sensible schedulers over $1 / \left(\frac{1}{recall^{\mathfrak{S}}(\mathsf{Cause})} - 1 \right)$ which is the same as taking the infimum of all sensible schedulers over $recall^{\mathfrak{S}}(\mathsf{Cause})$. This amounts to computing

$$\inf_{\mathfrak{S}} recall^{\mathfrak{S}}(\mathsf{Cause}) = \mathrm{Pr}_{\mathcal{M}}^{\min}(\, \Diamond\mathsf{Cause} \mid \Diamond\mathsf{Eff}\,),$$

which can be computed in polynomial time by [BKKM14, Mär20]. $\qquad\square$

In contrast to these results, we are not aware of known concepts that are applicable for computing the f-score. Indeed, this quality measure is efficiently computable:

Theorem 5.5. *The value fscore*(Cause) *and corresponding worst-case schedulers are computable in polynomial time.*

The remainder of this subsection is devoted to the proof of Theorem 5.5. We can express *fscore*(Cause) in terms of the supremum of a quotient of reachability probabilities for disjoint sets of terminal states. More precisely, under assumptions (A1)-(A3) and assuming *fscore*(Cause) > 0, we have:

$$fscore(\mathsf{Cause}) = \frac{2}{X+2} \quad \text{where} \quad X = \sup_{\mathfrak{S}} \frac{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{noeff}_{\mathsf{fp}}) + \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{unc}})}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{cov}})}$$

where $\mathfrak{S}$ ranges over all schedulers with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{cov}}) > 0$. Moreover, we can show that we can handle the corner case of *fscore*(Cause) $= 0$.

Lemma 5.6. *Let* Cause *be an SPR or a GPR cause. Then, the following three statements are equivalent:*

(a) *recall*(Cause) $= 0$
(b) *fscore*(Cause) $= 0$
(c) *There is a scheduler* $\mathfrak{S}$ *such that* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Eff}) > 0$ *and* $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{Cause}) = 0$.

Proof. Let C = Cause. Using results of [BKKM14, Mär20], there exist schedulers $\mathfrak{T}$ and $\mathfrak{U}$ with

- $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{Eff}) > 0$ and $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\, \Diamond\mathsf{C} \mid \Diamond\mathsf{Eff}\,) = \inf_{\mathfrak{S}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond\mathsf{C} \mid \Diamond\mathsf{Eff}\,)$ where $\mathfrak{S}$ ranges over all schedulers with positive effect probability,
- $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\Diamond\mathsf{C}) > 0$ and $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\, \Diamond\mathsf{Eff} \mid \Diamond\mathsf{C}\,) = \inf_{\mathfrak{S}} \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\, \Diamond\mathsf{Eff} \mid \Diamond\mathsf{C}\,)$ where $\mathfrak{S}$ ranges over all schedulers with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{C}) > 0$.

In particular, *recall*(C) $= \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\, \Diamond\mathsf{C} \mid \Diamond\mathsf{Eff}\,)$ and *precision*(C) $= \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{U}}(\, \Diamond\mathsf{Eff} \mid \Diamond\mathsf{C}\,)$. By (GPR) applied to $\mathfrak{U}$ and $\mathfrak{T}$ (recall that each SPR cause is a GPR cause too, see Lemma 3.3), we obtain the following statements (i) and (ii):

(i) : $\mathrm{p} \stackrel{\mathrm{def}}{=} precision(\mathsf{C}) > 0$
(ii) : If $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{C}) > 0$ then $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{C} \wedge \Diamond\mathsf{Eff}) > 0$ and therefore *recall*(C) > 0.

Obviously, if there is no scheduler $\mathfrak{S}$ as in statement (c) then $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond\mathsf{C}) > 0$. Thus, from (ii) we get:

(iii) : If there is no scheduler $\mathfrak{S}$ as in statement (c) then *recall*(C) > 0.

"(a) $\implies$ (b)": We prove *fscore*(C) > 0 implies *recall*(C) > 0. If *fscore*(C) > 0 then, by definition of the f-score, there is no scheduler $\mathfrak{S}$ as in statement (c). But then *recall*(C) > 0 by statement (iii).

"(b) $\implies$ (c)": Let *fscore*(C) $= 0$. Suppose by contradiction that there is no scheduler as in (c). Again by (iii) we obtain *recall*(C) > 0. But then, for each scheduler $\mathfrak{S}$ with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{C}) > 0$:

$$precision^{\mathfrak{S}}(\mathsf{C}) \geqslant \mathrm{p} \stackrel{(\mathrm{i})}{>} 0$$

and, with $r \stackrel{\text{def}}{=} recall(C)$:

$$recall^{\mathfrak{S}}(C) \geqslant r > 0$$

The harmonic mean as a function $]0,1]^2 \to \mathbb{R}$, $(x,y) \mapsto 2\frac{xy}{x+y}$ is monotonically increasing in both arguments. But then:

$$fscore^{\mathfrak{S}}(C) \geqslant 2\frac{p \cdot r}{p+r} > 0$$

Hence, $fscore(C) = \inf_{\mathfrak{S}} fscore^{\mathfrak{S}}(C) \geqslant 2\frac{p \cdot r}{p+r} > 0$. Contradiction.

"(c) $\Longrightarrow$ (a)": Let $\mathfrak{S}$ be a scheduler as in statement (c). Then,

$$\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond C \mid \Diamond \mathsf{Eff}) = 0.$$

Hence: $recall(C) = \Pr_{\mathcal{M}}^{\min}(\Diamond C \mid \Diamond \mathsf{Eff}) = 0.$ $\qquad\square$

The remaining task to prove Theorem 5.5 is a generally applicable technique for computing extremal ratios of reachability probabilities in MDPs without ECs.

Max/min ratios of reachability probabilities for disjoint sets of terminal states. Suppose we are given an MDP $\mathcal{M} = (S, Act, P, \mathsf{init})$ without ECs and disjoint subsets $U, V \subseteq S$ of terminal states. Given a scheduler $\mathfrak{S}$ with $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V) > 0$ we define:

$$ratio_{\mathcal{M}}^{\mathfrak{S}}(U,V) = \frac{\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond U)}{\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V)}$$

The goal is an algorithm for computing the extremal values:

$$ratio_{\mathcal{M}}^{\min}(U,V) = \inf_{\mathfrak{S}} ratio_{\mathcal{M}}^{\mathfrak{S}}(U,V) \quad \text{and} \quad ratio_{\mathcal{M}}^{\max}(U,V) = \sup_{\mathfrak{S}} ratio_{\mathcal{M}}^{\mathfrak{S}}(U,V)$$

where $\mathfrak{S}$ ranges over all schedulers with $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V) > 0$.

To compute these, we rely on a polynomial reduction to the classical *stochastic shortest path problem* [BT91]. For this, consider the MDP $\mathcal{N}$ arising from $\mathcal{M}$ by adding reset transitions from all terminal states $t \in S \backslash V$ to init. Thus, exactly the V-states are terminal in $\mathcal{N}$. $\mathcal{N}$ might contain ECs, which, however, do not intersect with V. We equip $\mathcal{N}$ with the weight function that assigns 1 to all states in U and 0 to all other states. For a scheduler $\mathfrak{T}$ with $\Pr_{\mathcal{N}}^{\mathfrak{T}}(\Diamond V) = 1$, let $E_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V)$ be the expected accumulated weight until reaching V under $\mathfrak{T}$. Let $E_{\mathcal{N}}^{\min}(\boxplus V) = \inf_{\mathfrak{T}} E_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V)$ and $E_{\mathcal{N}}^{\max}(\boxplus V) = \sup_{\mathfrak{T}} E_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V)$, where $\mathfrak{T}$ ranges over all schedulers with $\Pr_{\mathcal{N}}^{\mathfrak{T}}(\Diamond V) = 1$. We can rely on known results [BT91, dA99, BBD$^+$18] to obtain that both $E_{\mathcal{N}}^{\min}(\boxplus V)$ and $E_{\mathcal{N}}^{\max}(\boxplus V)$ are computable in polynomial time. As $\mathcal{N}$ has only non-negative weights, $E_{\mathcal{N}}^{\min}(\boxplus V)$ is finite and a corresponding MD-scheduler with minimal expectation exists. If $\mathcal{N}$ has an EC containing at least one U-state, which is the case iff $\mathcal{M}$ has a scheduler $\mathfrak{S}$ with $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond U) > 0$ and $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V) = 0$, then $E_{\mathcal{N}}^{\max}(\boxplus V) = +\infty$. Otherwise, $E_{\mathcal{N}}^{\max}(\boxplus V)$ is finite and the maximum is achieved by an MD-scheduler as well.

Theorem 5.7. *Let $\mathcal{M}$ be an MDP without ECs and U, V disjoint sets of terminal states in $\mathcal{M}$, and let $\mathcal{N}$ be as before. Then, $ratio_{\mathcal{M}}^{\min}(U,V) = E_{\mathcal{N}}^{\min}(\boxplus V)$ and $ratio_{\mathcal{M}}^{\max}(U,V) = E_{\mathcal{N}}^{\max}(\boxplus V)$. Thus, both values are computable in polynomial time, and there is an MD-scheduler minimizing $ratio_{\mathcal{M}}^{\mathfrak{S}}(U,V)$, and an MD-scheduler maximizing $ratio_{\mathcal{M}}^{\mathfrak{S}}(U,V)$ if $ratio_{\mathcal{M}}^{\max}(U,V)$ is finite.*

Example 5.8. Consider the MDP $\mathcal{M}$ from Figure 14 with $\mathsf{Eff} = \{\mathsf{eff}_{\mathsf{unc}}, \mathsf{eff}_{\mathsf{cov}}\}$ and suppose the task is to compute $covrat(c) = ratio_{\mathcal{M}}^{\min}(\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}})$. The construction for the algorithm is depicted in Figure 15 resulting in $\mathcal{N}$, where reset transitions for noeff and $\mathsf{eff}_{\mathsf{cov}}$ have been added (red edges) and $\mathsf{eff}_{\mathsf{unc}}$ is the only terminal state. The weight function now assigns 1 to $\mathsf{eff}_{\mathsf{cov}}$ and 0 to all others. By Theorem 5.7 we have $covrat(c) = ratio_{\mathcal{M}}^{\min}(\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}) = E_{\mathcal{N}}^{\min}(\boxplus \mathsf{eff}_{\mathsf{unc}})$. $\qquad\triangleleft$
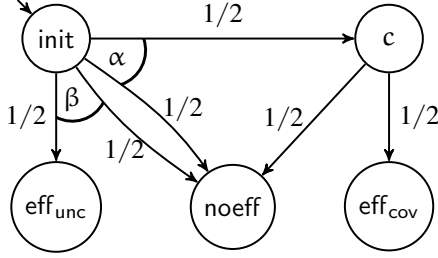
Figure 14: MDP $\mathcal{M}$ from Example 5.8
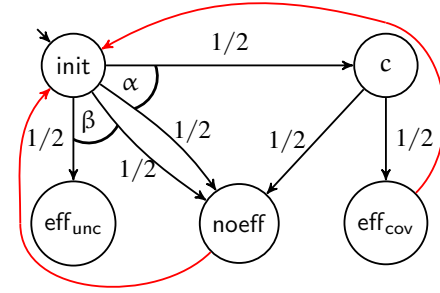
Figure 15: MDP $\mathcal{N}$ with reset transitions
for $ratio_{\mathcal{M}}^{\min}(\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}})$

Proof of Theorem 5.7. $\mathcal{M}$ has a scheduler $\mathfrak{S}$ with $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond U) > 0$ and $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V) = 0$ if and only if the transformed MDP $\mathcal{N}$ in Section 5.2 (Max/min ratios of reachability probabilities for disjoint sets of terminal states) has an EC containing at least one U-state. Therefore, we then have $\mathrm{E}_{\mathcal{N}}^{\max}(\boxplus V) = +\infty$. Otherwise, $\mathrm{E}_{\mathcal{N}}^{\max}(\boxplus V)$ is finite.

For the following we only consider $ratio_{\mathcal{M}}^{\min}(U, V) = \mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$ since the arguments for the maximum are similar. First, we show $ratio_{\mathcal{M}}^{\min}(U, V) \geqslant \mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$. For this, we consider an arbitrary scheduler $\mathfrak{S}$ for $\mathcal{M}$. Let

$$x = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond U) \qquad\qquad p = \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V) \qquad\qquad q = 1 - x - p$$

For $p > 0$ we have

$$\frac{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond U)}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond V)} = \frac{x}{p}$$

Let $\mathfrak{T}$ be the scheduler that behaves as $\mathfrak{S}$ in the first round and after each reset. Then:

$$\mathrm{E}_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V) = \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} n \cdot x^n \cdot \binom{n+k}{k} q^k \cdot p \stackrel{(5.2)}{=} \frac{x}{p} \tag{5.1}$$

where (5.2) relies on some basic calculations (see Lemma 5.9). This yields:

$$ratio_{\mathcal{M}}^{\mathfrak{S}}(U, V) = \frac{x}{p} = \mathrm{E}_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V) \geqslant \mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$$

Hence, $ratio_{\mathcal{M}}^{\min}(U, V) \geqslant \mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$.

For the other direction $\mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V) \geqslant ratio_{\mathcal{M}}^{\min}(U, V)$, we use the fact that there is an MD-scheduler $\mathfrak{T}$ for $\mathcal{N}$ such that $\mathrm{E}_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V) = \mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$. $\mathfrak{T}$ can be viewed as an MD-scheduler for the original MDP $\mathcal{M}$. Again we can rely on (5.1) to obtain that:

$$\mathrm{E}_{\mathcal{N}}^{\mathfrak{T}}(\boxplus V) = \frac{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond U)}{\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{T}}(\Diamond V)} = ratio_{\mathcal{M}}^{\mathfrak{T}}(U, V) \geqslant ratio_{\mathcal{M}}^{\min}(U, V)$$

But this yields $\mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V) \geqslant ratio_{\mathcal{M}}^{\min}(U, V)$. For $\mathrm{E}_{\mathcal{N}}^{\max}(\boxplus V) = ratio_{\mathcal{M}}^{\max}(U, V)$ we use similar arguments. We can now rely on known results [BT91, dA99, BBD$^+$18] to compute $\mathrm{E}_{\mathcal{N}}^{\min}(\boxplus V)$ and $\mathrm{E}_{\mathcal{N}}^{\max}(\boxplus V)$ in polynomial time. $\qquad\square$

Lemma 5.9. *Let* $x, p, q \in [0,1]$ *such that* $x+q+p = 1$. *Then:*

$$\sum_{n=0}^{\infty} \sum_{k=0}^{\infty} n \cdot x^n \cdot \binom{n+k}{k} q^k \cdot p = \frac{x}{p} \tag{5.2}$$

Proof. We first show for $0 < q < 1$, $n \in \mathbb{N}$ and

$$a_n \overset{\text{def}}{=} \sum_{k=0}^{\infty} \binom{n+k}{k} q^k,$$

we have

$$a_n = \frac{1}{(1-q)^{n+1}}$$

This is done by induction on n. The claim is clear for $n=0$. For the step of induction we use:

$$\binom{n+1+k}{k} = \binom{n+k}{k} + \binom{n+k}{k-1} = \binom{n+k}{k} + \binom{(n+1)+(k-1)}{k-1}$$

But this yields $a_{n+1} = a_n + q \cdot a_{n+1}$. Hence:

$$a_{n+1} = \frac{a_n}{1-q}$$

The claim then follows directly from the induction hypothesis.

The statement of Lemma 5.9 now follows by some calculations and the preliminary induction.

$$\sum_{n=0}^{\infty} \sum_{k=0}^{\infty} n \cdot x^n \cdot \binom{n+k}{k} q^k \cdot p = \sum_{n=0}^{\infty} n \cdot x^n \cdot \frac{1}{(1-q)^{n+1}} \cdot p$$

$$= \frac{p}{1-q} \cdot \sum_{n=0}^{\infty} n \cdot \left(\frac{x}{1-q}\right)^n$$

$$= \frac{p}{1-q} \cdot \frac{\frac{x}{1-q}}{\left(1 - \frac{x}{1-q}\right)^2}$$

$$= \frac{px}{(1-q-x)^2} = \frac{px}{p^2} = \frac{x}{p}$$

where we use $p = 1-q-x$. $\qquad\square$

Applying this framework for $ratio_{\mathcal{M}}^{\max}(U, V)$ to the f-score we now prove Theorem 5.5.

Proof of Theorem 5.5. We use the simplifying assumptions (A1)-(A3) that can be made due to Lemmas 5.2 and 5.3. For *fscore*(Cause) we have after some straight-forward transformations

$$fscore^{\mathfrak{S}}(\mathsf{Cause}) = \frac{2tp^{\mathfrak{S}}}{2tp^{\mathfrak{S}} + fn^{\mathfrak{S}} + fp^{\mathfrak{S}}}.$$

Using this we get

$$\frac{2}{fscore^{\mathfrak{S}}(\mathsf{Cause})} - 2 = \frac{fp^{\mathfrak{S}} + fn^{\mathfrak{S}}}{tp^{\mathfrak{S}}} = \frac{Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{noeff}_{\mathsf{fp}}) + Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{unc}})}{Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\mathsf{eff}_{\mathsf{cov}})}$$

Thus, the task is to compute

$$X = \sup_{\mathfrak{S}} \frac{2}{fscore^{\mathfrak{S}}(\mathsf{Cause})} - 2 = \sup_{\mathfrak{S}} \frac{\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{noeff}_{\mathsf{fp}}) + \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{eff}_{\mathsf{unc}})}{\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{eff}_{\mathsf{cov}})},$$

where $\mathfrak{S}$ ranges over all schedulers with $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{eff}_{\mathsf{cov}}) > 0$. We have

$$fscore(\mathsf{Cause}) = \frac{2}{X+2}.$$

But X can be expressed as a supremum in the form of Theorem 5.7. This yields the claim that the optimal value is computable in polynomial time.

In case $fscore(\mathsf{Cause}) = 0$, we do not obtain an optimal scheduler via Theorem 5.7. Lemma 5.6, however, shows that there is a scheduler $\mathfrak{S}$ with $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) > 0$ and $\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Cause}) = 0$. Such a scheduler can be computed in polynomial time as any (memoryless) scheduler in the largest sub-MDP of $\mathcal{M}$ that does not contain states in Cause. (This sub-MDP can be constructed by successively removing states and state-action pairs.) □

5.3. Quality-optimal probability-raising causes.
For the computation there is no difference between GPR and SPR causes as only the quality properties of the set are in question. However, when finding optimal causes the distinction makes a difference. Here, we say an SPR cause Cause is *recall-optimal* if $recall(\mathsf{Cause}) = \max_C recall(C)$ where C ranges over all SPR causes. Likewise, *ratio-optimality* resp. *f-score-optimality* of Cause means maximality of $covrat(\mathsf{Cause})$ resp. $fscore(\mathsf{Cause})$ among all SPR causes. Recall-, ratio- and f-score-optimality for GPR causes are defined accordingly.

Lemma 5.10. *Let* Cause *be an SPR or a GPR cause. Then,* Cause *is recall-optimal if and only if* Cause *is ratio-optimal.*

Proof. Essentially the proof uses the same connection between *recall* and *covrat* as Corollary 5.4. Here we do not assume (A1)-(A3). However, for each scheduler $\mathfrak{S}$ and each set C of states we have:

$$\Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \mathsf{Eff}) = \mathsf{fn}_C^{\mathfrak{S}} + \mathsf{tp}_C^{\mathfrak{S}}$$

where $\mathsf{fn}_C^{\mathfrak{S}} = \Pr_{\mathcal{M}}^{\mathfrak{S}}((\neg\Diamond C) \wedge \Diamond \mathsf{Eff})$ and $\mathsf{tp}_C^{\mathfrak{S}} = \Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond(C \wedge \Diamond \mathsf{Eff}))$. If C is a cause where $\mathsf{fn}_C^{\mathfrak{S}}$ is positive then $covrat^{\mathfrak{S}}(C) = \frac{\mathsf{tp}_C^{\mathfrak{S}}}{\mathsf{fn}_C^{\mathfrak{S}}}$ and $recall^{\mathfrak{S}}(C) = \frac{\mathsf{tp}_C^{\mathfrak{S}}}{\mathsf{fn}_C^{\mathfrak{S}} + \mathsf{tp}_C^{\mathfrak{S}}}$.

For all non-negative reals p, q, p', q' where $q, q' > 0$ we have:

$$\frac{p}{q} < \frac{p'}{q'} \qquad \text{iff} \qquad \frac{p}{p+q} < \frac{p'}{p'+q'}.$$

Hence, if C is fixed and $\mathfrak{S}$ ranges over all schedulers with $\mathsf{tp}_C^{\mathfrak{S}} > 0$:

$$\frac{\mathsf{tp}_C^{\mathfrak{S}}}{\mathsf{fn}_C^{\mathfrak{S}}} \text{ is minimal iff } \frac{\mathsf{tp}_C^{\mathfrak{S}}}{\mathsf{fn}_C^{\mathfrak{S}} + \mathsf{tp}_C^{\mathfrak{S}}} \text{ is minimal}$$

Thus, if C is fixed and $\mathfrak{S} = \mathfrak{S}_C$ is a scheduler achieving the worst-case (i.e., minimal) coverage ratio for C then $\mathfrak{S}$ achieves the minimal recall for C, and vice versa.

Let now $\mathsf{fn}_C = \mathsf{fn}_C^{\mathfrak{S}_C}$, $\mathsf{tp}_C = \mathsf{tp}_C^{\mathfrak{S}_C}$ where $\mathfrak{S}_C$ is a scheduler that minimizes the coverage ratio and minimizes the recall for cause set C. Then:

$$covrat(C) = \frac{\mathsf{tp}_C}{\mathsf{fn}_C} \text{ is maximal iff } \frac{\mathsf{tp}_C}{\mathsf{fn}_C + \mathsf{tp}_C} \text{ is maximal iff } recall(C) \text{ is maximal}$$

where the extrema range over all SPR resp. GPR causes C. This yields the claim. □
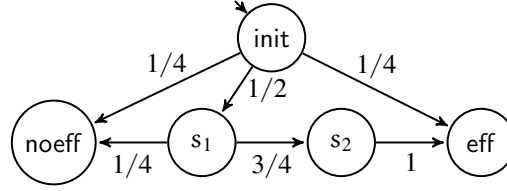
Figure 16: Markov chain from Remark 5.12

Recall- and ratio-optimal SPR causes. The techniques of Section 4.1 yield an algorithm for generating a canonical SPR cause with optimal recall and coverage ratio. To see this, let $\mathcal{C}$ denote the set of all states which constitute a singleton SPR cause. The canonical cause $\mathsf{CanCause}$ is defined as the set of states $c \in \mathcal{C}$ such that there is a scheduler $\mathfrak{S}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\mathcal{C})\,\mathsf{U}\,c) > 0$. So to speak $\mathsf{CanCause}$ is the "front" of $\mathcal{C}$. Obviously, $\mathcal{C}$ and $\mathsf{CanCause}$ are computable in polynomial time.

Theorem 5.11. *If $\mathcal{C} \neq \varnothing$ then* $\mathsf{CanCause}$ *is a ratio- and recall-optimal SPR cause.*

Proof. By definition of SPR causes any subset $C \subseteq \mathcal{C}$ satisfying $\mathrm{Pr}^{\mathrm{max}}(\neg C\,\mathsf{U}\,c)$ for each $c \in C$ constitutes an SPR cause and thus $\mathsf{CanCause}$ is also an SPR cause. Optimality is a consequence as $\mathsf{CanCause}$ even yields path-wise optimal coverage in the following sense. If C is any SPR cause then $C \subseteq \mathcal{C}$ by definition and for each path π in $\mathcal{M}$:

$$\pi \models (\neg\Diamond\mathsf{CanCause}) \wedge \Diamond\mathsf{Eff} \implies \pi \models (\neg\Diamond C) \wedge \Diamond\mathsf{Eff} \text{ and}$$
$$\pi \models \Diamond(C \wedge \Diamond\mathsf{Eff}) \implies \pi \models \Diamond(\mathsf{CanCause} \wedge \Diamond\mathsf{Eff}).$$

But then

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond(C \wedge \Diamond\mathsf{Eff})) \leqslant \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond(\mathsf{CanCause} \wedge \Diamond\mathsf{Eff})),$$
$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\Diamond C) \wedge \Diamond\mathsf{Eff}) \geqslant \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}((\neg\Diamond\mathsf{CanCause}) \wedge \mathsf{Eff})$$

for every scheduler $\mathfrak{S}$, which yields the claim. $\qquad\square$

Remark 5.12. It is not true that the canonical SPR cause $\mathsf{CanCause}$ is f-score-optimal. To see this, Consider the Markov chain from Figure 16. There we have $\mathsf{CanCause} = \{s_1\}$, which has *precision*($\mathsf{CanCause}$) $= \frac{3}{4}$ and *recall*($\mathsf{CanCause}$) $= \frac{3}{8}/(\frac{1}{4}+\frac{3}{8}) = \frac{3}{5}$. But the SPR cause $\{s_2\}$ has better f-score as its precision is 1 and it has the same recall as $\mathsf{CanCause}$. $\qquad\triangleleft$

F-score-optimal SPR cause. From Section 5.2, we see that f-score-optimal SPR causes in MDPs can be computed in polynomial space by computing the f-score for all potential SPR causes one by one in polynomial time (Theorem 5.5). As the space can be reused after each computation, this results in polynomial space. For Markov chains, we can do better and compute an f-score-optimal SPR cause in polynomial time via a polynomial reduction to the stochastic shortest path problem:

Theorem 5.13. *In Markov chains that have SPR causes, an f-score-optimal SPR cause can be computed in polynomial time.*

Proof. We regard the given Markov chain $\mathcal{M}$ as an MDP with a singleton action set $Act = \{\alpha\}$. As $\mathcal{M}$ has SPR causes, the set $\mathcal{C}$ of states that constitute a singleton SPR cause is nonempty. We may assume that $\mathcal{M}$ has no non-trivial (i.e., cyclic) bottom strongly connected components as we may collapse them. Let $w_c = \mathrm{Pr}_{\mathcal{M},c}(\Diamond\mathsf{Eff})$.

We switch from $\mathcal{M}$ to a new MDP $\mathcal{K}$ with state space $S_{\mathcal{K}} = S \cup \{\mathsf{eff}_{\mathsf{cov}}, \mathsf{noeff}_{\mathsf{fp}}\}$ with fresh states $\mathsf{noeff}_{\mathsf{fp}}$ and $\mathsf{eff}_{\mathsf{cov}}$ and the action set $Act_{\mathcal{K}} = \{\alpha, \gamma\}$. The MDP $\mathcal{K}$ arises from $\mathcal{M}$ by adding

(i) : for each SPR state $c \in \mathcal{C}$ a fresh state-action pair (c, γ) such that $P_{\mathcal{K}}(c, \gamma, \text{eff}_{\text{cov}}) = w_c$ and $P_{\mathcal{K}}(c, \gamma, \text{noeff}_{\text{fp}}) = 1 - w_c$ and

(ii) : reset transitions to init with action label α from the new state noeff_{fp} and all terminal states of $\mathcal{M}$, i.e., $P_{\mathcal{K}}(\text{noeff}_{\text{fp}}, \alpha, \text{init}) = 1$ and $P_{\mathcal{K}}(s, \alpha, \text{init}) = 1$ for $s \in \text{Eff}$ or if s is a terminal non-effect state of $\mathcal{M}$.

So, exactly eff_{cov} is terminal in $\mathcal{K}$, and $Act_{\mathcal{K}}(c) = \{\alpha, \gamma\}$ for $c \in \mathcal{C}$, while $Act_{\mathcal{K}}(s) = \{\alpha\}$ for all other states s. Intuitively, taking action γ in state $c \in \mathcal{C}$ selects c to be a cause state. The states in Eff represent uncovered effects in $\mathcal{K}$, while eff_{cov} stands for covered effects.

We assign weight 1 to all states in $U = \text{Eff} \cup \{\text{noeff}_{\text{fp}}\}$ and weight 0 to all other states of $\mathcal{K}$. Let $V = \{\text{eff}_{\text{cov}}\}$. Then, $f = E_{\mathcal{K}}^{\min}(\boxplus V)$ and an MD-scheduler $\mathfrak{S}$ for $\mathcal{K}$ such that $E_{\mathcal{K}}^{\mathfrak{S}}(\boxplus V) = f$ are computable in polynomial time. Let $\mathcal{C}_{\gamma}$ denote the set of states $c \in \mathcal{C}$ where $\mathfrak{S}(c) = \gamma$ and let Cause be the set of states $c \in \mathcal{C}_{\gamma}$ where $\mathcal{M}$ has a path satisfying $(\neg \mathcal{C}_{\gamma}) \cup c$. Then, Cause is an SPR cause of $\mathcal{M}$. With arguments as in Section 5.2 we obtain $\textit{fscore}(\text{Cause}) = 2/(f+2)$. It remains to show that Cause is f-score-optimal. Let C be an arbitrary SPR cause. Then, $C \subseteq \mathcal{C}$. Let $\mathfrak{T}$ be the MD-scheduler for $\mathcal{K}$ that schedules γ in C and α for all other states of $\mathcal{K}$. Then, $\textit{fscore}(C) = 2/(f^{\mathfrak{T}}+2)$ where $f^{\mathfrak{T}} = E_{\mathcal{K}}^{\mathfrak{T}}(\boxplus V)$. Hence, $f \leqslant f^{\mathfrak{T}}$, which yields $\textit{fscore}(\text{Cause}) \geqslant \textit{fscore}(C)$. $\qquad \square$

The naïve adaption of the construction presented in the proof of Theorem 5.13 for MDPs would yield a stochastic game structure where the objective of one player is to minimize the expected accumulated weight until reaching a target state. Although algorithms for *stochastic shortest path (SSP) games* are known [PB99], they rely on assumptions on the game structure which would not be satisfied here. However, for the threshold problem *SPR-f-score* where inputs are an MDP $\mathcal{M}$, Eff and $\vartheta \in \mathbb{Q}_{\geqslant 0}$ and the task is to decide the existence of an SPR cause whose f-score exceeds ϑ, we can establish a polynomial reduction to SSP games, which yields an NP$\cap$coNP upper bound:

Theorem 5.14. *The decision problem SPR-f-score is in* NP$\cap$coNP.

Recall that for a given SPR cause C and scheduler $\mathfrak{S}$ we have

$$\textit{fscore}^{\mathfrak{S}}(C) > \vartheta \text{ iff } \frac{2\text{tp}^{\mathfrak{S}}}{2\text{tp}^{\mathfrak{S}} + \text{fp}^{\mathfrak{S}} + \text{fn}^{\mathfrak{S}}} > \vartheta.$$

In order to proof the upper bound of SPR-f-score we reformulate the condition of SPR-f-score.

Lemma 5.15. *Let $\mathcal{M} = (S, Act, P, \text{init})$ be an MDP with a set of terminal states* Eff*, let C be an SPR cause for* Eff *in $\mathcal{M}$, and let ϑ be a rational. Then, $\textit{fscore}(C) > \vartheta$ iff*

$$2(1-\vartheta)\text{tp}^{\mathfrak{S}} - \vartheta\text{fp}^{\mathfrak{S}} - \vartheta\text{fn}^{\mathfrak{S}} > 0 \qquad\qquad (5.3)$$

for all schedulers $\mathfrak{S}$ for $\mathcal{M}$ with $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\text{Eff}) > 0$.

Proof. Assume that $\textit{fscore}(C) > \vartheta$ and let $\mathfrak{S}$ be a scheduler with $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\text{Eff}) > 0$. If $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond C) = 0$, then $\textit{fscore}(C)$ would be 0. So, $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond C) > 0$. Then,

$$\textit{fscore}^{\mathfrak{S}}(C) = \frac{2\text{tp}^{\mathfrak{S}}}{2\text{tp}^{\mathfrak{S}} + \text{fp}^{\mathfrak{S}} + \text{fn}^{\mathfrak{S}}} > \vartheta, \quad \text{which implies} \quad 2(1-\vartheta)\text{tp}^{\mathfrak{S}} - \vartheta\text{fp}^{\mathfrak{S}} - \vartheta\text{fn}^{\mathfrak{S}} > 0$$

Now, suppose that (5.3) holds for a scheduler $\mathfrak{S}$ with $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond\text{Eff}) > 0$. Let $\mathfrak{S}$ be a scheduler that minimizes $\textit{fscore}^{\mathfrak{S}}(C)$. Such a scheduler exists by Theorem 5.5. From (5.3), we conclude

$$2(1-\vartheta)\text{tp}^{\mathfrak{S}} - \vartheta\text{fp}^{\mathfrak{S}} - \vartheta\text{fn}^{\mathfrak{S}} > 0$$

and hence that $\textit{fscore}^{\mathfrak{S}}(C) > \vartheta$ as above. $\qquad\qquad \square$

Proof of Theorem 5.14. Let $\mathcal{M} = (S, Act, P, \text{init})$ be an MDP, $\text{Eff} \subseteq S$ a set of terminal states, and ϑ a rational. Consider $\mathcal{C}$, the set of states $c \in S \setminus \text{Eff}$ where $\{c\}$ is an SPR cause. If $\mathcal{C}$ is empty then the threshold problem is trivial as there is no SPR cause at all. Thus, we suppose that $\mathcal{C}$ is nonempty.

Note that $\text{Pr}_{\mathcal{M},c}^{\min}(\Diamond \text{Eff}) > 0$ for all $c \in \mathcal{C}$. As the states in Eff are not part of any end component of $\mathcal{M}$, no state $c \in \mathcal{C}$ is contained in an end component of $\mathcal{M}$ either. Let $\mathcal{N} = (S_{\mathcal{N}}, Act_{\mathcal{N}}, P_{\mathcal{N}}, \text{init}_{\mathcal{N}})$ be the MEC-quotient of $\mathcal{M}$ with the new additional terminal state $\bot$. The MEC-quotient $\mathcal{N}$ contains the states from Eff and $\mathcal{C}$.

Claim 1: There is an SPR cause C for Eff in $\mathcal{M}$ with *fscore*(C) $> \vartheta$ if and only if there is an SPR cause C$'$ for Eff in $\mathcal{N}$ with *fscore*(C$'$) $> \vartheta$.
Proof of Claim 1. We first observe that all reachability probabilities involved in the claim do not depend on the behavior during the traversal of MECs. Furthermore, staying inside a MEC in $\mathcal{M}$ can be mimicked in $\mathcal{N}$ by moving to $\bot$, and vice versa. More precisely, let $C \subseteq \mathcal{C}$. Then, analogously to Lemma 4.13, for each scheduler $\mathfrak{S}$ for $\mathcal{M}$, there is a scheduler $\mathfrak{T}$ for $\mathcal{N}$, and vice versa, such that

- $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Eff} \mid (\neg C) \cup c) = \text{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond \text{Eff} \mid (\neg C) \cup c)$ for all $c \in C$ for which the values are defined,
- $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Eff}) = \text{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond \text{Eff})$,
- $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond \text{Eff} \mid \Diamond C) = \text{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond \text{Eff} \mid \Diamond C)$ if the values are defined, and
- $\text{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\Diamond C \mid \Diamond \text{Eff}) = \text{Pr}_{\mathcal{N}}^{\mathfrak{T}}(\Diamond C \mid \Diamond \text{Eff})$ if the values are defined.

Hence, C is an SPR cause for Eff in $\mathcal{M}$ if and only if it is in $\mathcal{N}$ and furthermore, if it is an SPR cause, the f-score of C in $\mathcal{M}$ and in $\mathcal{N}$ agree. This finishes the proof of Claim 1.

Model transformation for ensuring positive effect probabilities. Recall that the f-score is only defined for schedulers reaching Eff with positive probability. Now, we will provide a further model transformation that will ensure that Eff is reached with positive probability under all schedulers. If this is already the case, there is nothing to do. So, we assume now that $\text{Pr}_{\mathcal{N},\text{init}_{\mathcal{N}}}^{\min}(\Diamond \text{Eff}) = 0$.

We define the subset of states from which Eff can be avoided as $D \subseteq S_{\mathcal{N}}$ by

$$D \overset{\text{def}}{=} \{s \in S_{\mathcal{N}} \mid \text{Pr}_{\mathcal{N},s}^{\min}(\Diamond \text{Eff}) = 0\}.$$

Note that $\text{init}_{\mathcal{N}} \in D$. For each $s \in D$, we further define the set of actions minimizing the reachability probability of Eff from s by

$$Act^{\min}(s) = \{\alpha \in Act_{\mathcal{N}}(s) \mid P_{\mathcal{N}}(s, \alpha, D) = 1\}.$$

Finally, let $E \subseteq D$ be the set of states that are reachable from $\text{init}_{\mathcal{N}}$ when only choosing actions from $Act^{\min}(\cdot)$. Note that E does not contain any states from $\mathcal{C}$, meaning no state in E constitutes a singleton SPR cause.

All schedulers that reach Eff with positive probability in $\mathcal{N}$ have to leave the sub-MDP consisting of E and the actions in $Act^{\min}(\cdot)$ at some point. Let us call this sub-MDP $\mathcal{N}_E^{\min}$. We define the set of state-action pairs Π that leave the sub-MDP $\mathcal{N}_E^{\min}$:

$$\Pi \overset{\text{def}}{=} \{(s, \alpha) \mid s \in E \text{ and } \alpha \in Act_{\mathcal{N}}(s) \setminus Act^{\min}(s)\}.$$

We now construct a further MDP $\mathcal{K}$. The idea is that $\mathcal{K}$ behaves like the end-component free MDP $\mathcal{N}$ after initially a scheduler is forced to choose a probability distribution over state-action pairs from Π. In this way, Eff is reached with positive probability under all schedulers. Given an SPR cause, we will observe that for the f-score of this cause under a scheduler, it is only important how large the probabilities with which state action pairs from Π are chosen are relative to each other while the absolute values are not important. Due to this observation, for each SPR cause C and for each

scheduler $\mathfrak{S}$ for $\mathcal{N}$ that reaches Eff with positive probability, we can then construct a scheduler for $\mathcal{K}$ that leads to the same recall and precision of C.

Formally, $\mathcal{K}$ is defined as follows: The state space is $S_{\mathcal{N}} \cup \{\text{init}_{\mathcal{K}}\}$ where $\text{init}_{\mathcal{K}}$ is a fresh initial state. For all states in $S_{\mathcal{N}}$, the same actions as in $\mathcal{N}$ are available with the same transition probabilities. I.e., for all $s, t \in S_{\mathcal{N}}$,

$$Act_{\mathcal{K}}(s) \stackrel{\text{def}}{=} Act_{\mathcal{N}}(s) \quad \text{and} \quad P_{\mathcal{K}}(s, \alpha, t) \stackrel{\text{def}}{=} P_{\mathcal{N}}(s, \alpha, t) \text{ for all } \alpha \in Act_{\mathcal{K}}(s).$$

For each state-action pair (s, α) from Π, we now add a new action $\beta_{(s,\alpha)}$ that is enabled only in $\text{init}_{\mathcal{K}}$. These are all actions enabled in $\text{init}_{\mathcal{K}}$, i.e.,

$$Act_{\mathcal{K}}(\text{init}_{\mathcal{K}}) \stackrel{\text{def}}{=} \{\beta_{(s,\alpha)} \mid (s, \alpha) \in \Pi\}.$$

For each state $t \in S_{\mathcal{N}}$, we define the transition probabilities under $\beta_{(s,\alpha)}$ by

$$P_{\mathcal{K}}(\text{init}_{\mathcal{K}}, \beta_{(s,\alpha)}, t) \stackrel{\text{def}}{=} P_{\mathcal{N}}(s, \alpha, t).$$

Claim 2: A subset $C \subseteq \mathcal{C}$ such that for all $c \in C : \text{Pr}_{\mathcal{N}}^{\max}(\neg C \, \mathsf{U} \, c) > 0$ is an SPR cause for Eff in $\mathcal{N}$ with $fscore(C) > \vartheta$ if and only if for all schedulers $\mathfrak{T}$ for $\mathcal{K}$, we have

$$2(1-\vartheta)\text{tp}_{\mathcal{K}}^{\mathfrak{T}} - \vartheta\text{fp}_{\mathcal{K}}^{\mathfrak{T}} - \vartheta\text{fn}_{\mathcal{K}}^{\mathfrak{T}} > 0. \tag{5.4}$$

Proof of Claim 2. We first prove the direction "$\Rightarrow$". So, let C be an SPR cause for Eff in end-component free MDP $\mathcal{N}$ with $fscore(C) > \vartheta$. As first observation we have that in order to prove (5.4) for all schedulers $\mathfrak{T}$ for $\mathcal{K}$, it suffices to consider schedulers $\mathfrak{T}$ that start with a deterministic choice for state $\text{init}_{\mathcal{K}}$ and then behave in an arbitrary way.

To see this, we consider the MDP $\mathcal{K}_C$ which consists of two copies of $\mathcal{K}$: "before C" and "after C". When $\mathcal{K}_C$ enters a C-state in the first copy ("before C"), it switches to the second copy ("after C") and stays there forever. Let us write $(s, 1)$ for state s in the first copy and $(s, 2)$ for the copy of state s in the second copy. Thus, in $\mathcal{K}_C$ the event $\Diamond C \wedge \Diamond Eff$ is equivalent to reaching a state $(eff, 2)$ where $eff \in Eff$, while $\Diamond C \wedge \neg \Diamond Eff$ is equivalent to reaching a non-terminal state in the second copy, while $\neg \Diamond C \wedge \Diamond Eff$ corresponds to the event reaching an effect state in the first copy.

Obviously, there is a one-to-one-correspondence of the schedulers of $\mathcal{K}$ and $\mathcal{K}_C$. As $\mathcal{K}$ has no end components so does $\mathcal{K}_C$. Therefore, a terminal state will be reached almost surely under every scheduler. Furthermore, we equip $\mathcal{K}_C$ with a weight function on states which assigns
– weight $2(1-\vartheta)$ to the states $(eff, 2)$ where $eff \in Eff$,
– weight $-\vartheta$ to the states $(eff, 1)$ where $eff \in Eff$ and to the states $(s, 2)$ where s is a terminal non-effect state in $\mathcal{K}$ (and $\mathcal{K}_C$), and
– weight 0 to all other states.
Let V denote the set of all terminal states in $\mathcal{K}_C$. Then, the expression on the left hand side of (5.4) equals $\text{E}_{\mathcal{K}_C}^{\mathfrak{T}}(\boxplus V)$, the expected accumulated weight until reaching a terminal state under scheduler $\mathfrak{T}$. Hence, (5.4) holds for all schedulers $\mathfrak{T}$ in $\mathcal{K}$ if and only if $\text{E}_{\mathcal{K}_C}^{\min}(\boxplus V) > 0$.

It is well-known that the minimal expected accumulated weight in EC-free MDPs is achieved by an MD-scheduler [BK08]. Thus, there is an MD-scheduler $\mathfrak{T}$ of $\mathcal{K}_C$ with $\text{E}_{\mathcal{K}_C}^{\min}(\boxplus V) = \text{E}_{\mathcal{K}_C}^{\mathfrak{T}}(\boxplus V)$. When viewed as a scheduler of $\mathcal{K}$, $\mathfrak{T}$ behaves memoryless deterministic before reaching C. In particular, $\mathfrak{T}$'s initial choice in $\text{init}_{\mathcal{K}}$ is deterministic.

Recall the set $Act^{\min}(s)$ of actions minimizing the reachability probability of Eff from s. Consider a scheduler $\mathfrak{T}$ for $\mathcal{K}$ with a deterministic choice $\mathfrak{T}(\text{init}_{\mathcal{K}})(\beta_{(s,\alpha)}) = 1$ where $(s, \alpha) \in \Pi$. To construct an analogous scheduler $\mathfrak{S}$ of $\mathcal{N}$, we pick an MD-scheduler $\mathfrak{U}$ of the sub-MDP $\mathcal{N}_E^{\min}$ of $\mathcal{N}$ induced by

the state-action pairs (u, β) where $u \in E$ and $\beta \in Act^{min}(u)$ such that there is a $\mathfrak{U}$-path from $init_{\mathcal{N}}$ to state s.

Scheduler $\mathfrak{S}$ of $\mathcal{N}$ operates with the mode $\mathfrak{m}_1$ and the modes $\mathfrak{m}_{2,t}$ for $t \in S_{\mathcal{N}}$. In its initial mode $\mathfrak{m}_1$, scheduler $\mathfrak{S}$ behaves as $\mathfrak{U}$ as long as state s has not been visited. When having reached state s in mode $\mathfrak{m}_1$, then $\mathfrak{S}$ schedules the action α with probability 1. Let $t \in S_{\mathcal{N}}$ be the state that $\mathfrak{S}$ reaches via the α-transition from s. Then, $\mathfrak{S}$ switches to mode $\mathfrak{m}_{2,t}$ and behaves from then on as the residual scheduler $res(\mathfrak{T}, \varpi)$ of $\mathfrak{T}$ for the $\mathfrak{T}$-path $\varpi = init_{\mathcal{K}} \beta_{(s,\alpha)} t$ in $\mathcal{K}$. That is, after having scheduled the action $\beta_{(s,\alpha)}$, scheduler $\mathfrak{S}$ behaves exactly as $\mathfrak{T}$.

Let λ denote $\mathfrak{S}$'s probability to leave mode $\mathfrak{m}_1$, which equals $\mathfrak{U}$'s probability to reach s from $init_{\mathcal{N}}$. Thus, $\lambda = Pr_{\mathcal{N}}^{\mathfrak{U}}(\Diamond s)$ when $\mathfrak{U}$ is viewed as a scheduler of $\mathcal{N}$. As E is disjoint from C and Eff, scheduler $\mathfrak{S}$ stays forever in mode $\mathfrak{m}_1$ and never reaches a state in $C \cup Eff$ with probability $1 - \lambda$.

$\mathfrak{S}$ and $\mathfrak{T}$ behave identically after choosing the state-action pair $(s, \alpha) \in \Pi$ or the corresponding action $\beta_{(s,\alpha)}$, respectively, which implies that

- $Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond C \wedge \Diamond Eff) = \lambda \cdot Pr_{\mathcal{K}}^{\mathfrak{T}}(\Diamond C \wedge \Diamond Eff)$,
- $Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond Eff) = \lambda \cdot Pr_{\mathcal{K}}^{\mathfrak{T}}(\Diamond Eff)$ and
- $Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond C \wedge \neg \Diamond Eff) = \lambda \cdot Pr_{\mathcal{K}}^{\mathfrak{T}}(\Diamond C \wedge \neg \Diamond Eff)$.

Recall the sub-MDP $\mathcal{N}_E^{min}$ consisting of E and the actions in $Act^{min}(\cdot)$. As $\mathfrak{S}$ leaves the sub-MDP $\mathcal{N}_E^{min}$ with probability $\lambda > 0$, we have $Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond Eff) > 0$. By Lemma 5.15, we can conclude that

$$2(1-\vartheta)tp_{\mathcal{N}}^{\mathfrak{S}} - \vartheta fp_{\mathcal{N}}^{\mathfrak{S}} - \vartheta fn_{\mathcal{N}}^{\mathfrak{S}} > 0.$$

By the equations above, this in turn implies that

$$2(1-\vartheta)tp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fn_{\mathcal{K}}^{\mathfrak{T}} > 0.$$

For the direction "$\Leftarrow$", first recall that any subset of $\mathcal{C}$ satisfying (M) is an SPR cause for Eff in $\mathcal{N}$ by definition of $\mathcal{C}$. Now, let $\mathfrak{S}$ be a scheduler for $\mathcal{N}$ with $Pr_{\mathcal{N}}^{\mathfrak{S}}(\Diamond Eff) > 0$. Let Γ be the set of finite $\mathfrak{S}$-paths γ in the sub-MDP $\mathcal{N}_E^{min}$ such that $\mathfrak{S}$ chooses an action in $Act_{\mathcal{N}}(last(\gamma)) \setminus Act^{min}(last(\gamma))$ with positive probability after γ where $last(\gamma)$ denotes the last state of γ. Let

$$q \stackrel{def}{=} \sum_{\gamma \in \Gamma} \sum_{\alpha \in Act_{\mathcal{N}}(last(\gamma)) \setminus Act^{min}(last(\gamma))} P_{\mathcal{N}}(\gamma) \cdot \mathfrak{S}(\gamma)(\alpha).$$

So, q is the overall probability that a state-action pair from Π is chosen under $\mathfrak{S}$. We now define a scheduler $\mathfrak{T}$ for $\mathcal{K}$: For each $\gamma \in \Gamma$ ending in a state s and each $\alpha \in Act_{\mathcal{N}}(s) \setminus Act^{min}(s)$, the scheduler $\mathfrak{T}$ chooses action $\beta_{(s,\alpha)}$ in $init_{\mathcal{K}}$ with probability $P_{\mathcal{N}}(\gamma) \cdot \mathfrak{S}(\gamma)(\alpha)/q$. When reaching a state t afterwards, $\mathfrak{T}$ behaves like $res(\mathfrak{S}, \gamma \alpha t)$ afterwards. Note that by definition this indeed defines a probability distribution over the actions in the initial state $init_{\mathcal{K}}$.

By assumption, we know that now

$$2(1-\vartheta)tp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fn_{\mathcal{K}}^{\mathfrak{T}} > 0.$$

As the probability with which an action $\beta_{(s,\alpha)}$ is chosen by $\mathfrak{T}$ for a $(s, \alpha) \in \Pi$ is $1/q$ times the probability that α is chosen in s to leave the sub-MDP $\mathcal{N}_E^{min}$ under $\mathfrak{S}$ in $\mathcal{N}$ and as the residual behavior is identical, we conclude that

$$2(1-\vartheta)tp_{\mathcal{N}}^{\mathfrak{S}} - \vartheta fp_{\mathcal{N}}^{\mathfrak{S}} - \vartheta fn_{\mathcal{N}}^{\mathfrak{S}} = q \cdot (2(1-\vartheta)tp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fp_{\mathcal{K}}^{\mathfrak{T}} - \vartheta fn_{\mathcal{K}}^{\mathfrak{T}}) > 0.$$

By Lemma 5.15, this shows that $fscore(C) > \vartheta$ in $\mathcal{N}$ and finishes the proof of Claim 2.

Construction of a game structure. Recall the set of singleton SPR causes $\mathcal{C}$. We now construct a *stochastic shortest path game* (see [PB99]) to check whether there is a subset $C \subseteq \mathcal{C}$ such that (5.4) holds in the EC-free MDP $\mathcal{K}$ in which visiting effect states always has a non-zero probability. Such a game is played on an MDP-like structure with the only difference that the set of states is partitioned into two sets indicating which player controls which states.

The game $\mathcal{G}$ has states $(S_{\mathcal{K}} \times \{yes, no\}) \cup \mathcal{C} \times \{choice\}$. On the subset $S_{\mathcal{K}} \times \{yes\}$, all actions and transition probabilities are just as in $\mathcal{K}$ and this copy of $\mathcal{K}$ cannot be left. Formally, for all $s, t \in S_{\mathcal{K}}$ and $\alpha \in Act_{\mathcal{K}}(s)$, we have $Act_{\mathcal{G}}((s, yes)) = Act_{\mathcal{K}}(s)$ and $P_{\mathcal{G}}((s, yes), \alpha, (t, yes)) = P_{\mathcal{K}}(s, \alpha, t)$.

In the "*no*"-copy, the game also behaves like $\mathcal{K}$ but when a state in $\mathcal{C}$ would be entered, the game moves to a state in $\mathcal{C} \times \{choice\}$ instead. In a state of the form $(c, choice)$ with $c \in \mathcal{C}$, two actions α and β are available. Choosing α leads to the state (c, yes) while choosing β leads to (c, no) with probability 1.

Formally, this means that for all state $s \in S_{\mathcal{K}}$, we define $Act_{\mathcal{G}}((s, no)) = Act_{\mathcal{K}}(s)$ and for all actions $\alpha \in Act_{\mathcal{K}}(s)$:

- $P_{\mathcal{G}}((s, no), \alpha, (t, no)) = P_{\mathcal{K}}(s, \alpha, t)$ for all states $t \in S_{\mathcal{K}} \setminus \mathcal{C}$
- $P_{\mathcal{G}}((s, no), \alpha, (c, choice)) = P_{\mathcal{K}}(s, \alpha, c)$ for all states $c \in \mathcal{C}$

For states $s \in S_{\mathcal{K}}$, $c \in \mathcal{C}$, and $\alpha \in Act_{\mathcal{K}}(s)$, we furthermore define:

$$P_{\mathcal{G}}((c, choice), \alpha, (c, yes)) = P_{\mathcal{G}}((c, choice), \beta, (c, no)) = 1.$$

Intuitively speaking, whether a state $c \in \mathcal{C}$ should belong to the cause set can be decided in the state $(c, choice)$. The "*yes*"-copy encodes that an effect state has been selected. More concretely, the "*yes*-copy" is entered as soon as α has been chosen in some state $(c, choice)$ and will never be left from then on. The "*no*"-copy of $\mathcal{K}$ then encodes that no state $c \in \mathcal{C}$ which has been selected to become a cause state has been visited so far. That is, if the current state of a play in $\mathcal{G}$ belongs to the *no*-copy then in all previous decisions in the states $(c, choice)$, action β has been chosen.

Finally, we equip the game with a weight structure. All states in $\mathsf{Eff} \times \{yes\}$ get weight $2(1-\vartheta)$. All remaining terminal states in $S_{\mathcal{K}} \times \{yes\}$ get weight $-\vartheta$. All states in $\mathsf{Eff} \times \{no\}$ get weight $-\vartheta$. All remaining states have weight 0.

The game is played between two players 0 and 1. Player 0 controls all states in $\mathcal{C} \times \{choice\}$ while player 1 controls the remaining states. The goal of player 0 is to ensure that the expected accumulated weight is > 0.

Claim 3: Player 0 has a winning strategy ensuring that the expected accumulated weight is > 0 in the game $\mathcal{G}$ if and only if there is a subset $C \subseteq \mathcal{C}$ in $\mathcal{K}$ which satisfies for all $c \in C : \mathrm{Pr}_{\mathcal{K}}^{\max}(\neg C \, \mathsf{U} \, c) > 0$ and for all schedulers $\mathfrak{T}$ for $\mathcal{K}$ we have

$$2(1-\vartheta)\mathsf{tp}_{\mathcal{K}}^{\mathfrak{T}} - \vartheta\mathsf{fp}_{\mathcal{K}}^{\mathfrak{T}} - \vartheta\mathsf{fn}_{\mathcal{K}}^{\mathfrak{T}} > 0. \tag{5.5}$$

Proof of Claim 3. As $\mathcal{K}$ has no end components, also in the game $\mathcal{G}$ a terminal state is reached almost surely under any pair of strategies. Hence, we can rely on the results of [PB99] that state that both players have an optimal memoryless deterministic strategy.

We start by proving direction "$\Rightarrow$" of Claim 3. Let ζ be a memoryless deterministic winning strategy for player 0. I.e., ζ assigns to each state in $\mathcal{C} \times \{choice\}$ an action from $\{\alpha, \beta\}$. We define

$$\mathcal{C}_{\alpha} \overset{\text{def}}{=} \{c \in \mathcal{C} \mid \zeta((c, choice)) = \alpha\}.$$

Note that $\mathcal{C}_{\alpha}$ is not empty as otherwise a positive expected accumulated weight in the game is not possible. (Here we use the fact that only the effect states in the *yes*-copy have positive weight and

that the *yes*-copy can only be entered by taking α in one of the states $(c, choice)$.) To ensure for all $c \in C : \Pr^{\max}_{\mathcal{K}}(\neg C \cup c) > 0$, we remove states that cannot be visited as the first state of this set:

$$C \stackrel{\text{def}}{=} \{c \in \mathcal{C}_\alpha \mid \mathcal{K}, c \models \exists (\neg \mathcal{C}_\alpha) \cup c\}.$$

Note that the strategies for player 0 in $\mathcal{G}$ which correspond to the sets $\mathcal{C}_\alpha$ and C lead to exactly the same plays.

Let $\mathfrak{T}$ be a scheduler for $\mathcal{K}$. This scheduler can be used as a strategy for player 1 in $\mathcal{G}$. Let us denote the expected accumulated weight when player 0 plays according to ζ and player 1 plays according to $\mathfrak{T}$ by $w(\zeta, \mathfrak{T})$. As ζ is winning for player 0 we have $w(\zeta, \mathfrak{T}) > 0$. By the construction of the game, it follows directly that

$$w(\zeta, \mathfrak{T}) = 2(1-\vartheta)\mathsf{tp}^{\mathfrak{T}}_{\mathcal{K}} - \vartheta \mathsf{fp}^{\mathfrak{T}}_{\mathcal{K}} - \vartheta \mathsf{fn}^{\mathfrak{T}}_{\mathcal{K}}.$$

Putting things together yields:

$$2(1-\vartheta)\mathsf{tp}^{\mathfrak{T}}_{\mathcal{K}} - \vartheta \mathsf{fp}^{\mathfrak{T}}_{\mathcal{K}} - \vartheta \mathsf{fn}^{\mathfrak{T}}_{\mathcal{K}} > 0 \tag{5.6}$$

For the other direction, suppose there is a set $C \subseteq \mathcal{C}$ that satisfies $\Pr^{\max}_{\mathcal{K}}(\neg C \cup c) > 0$ for all $c \in C$ and (5.5) for all schedulers $\mathfrak{T}$ for $\mathcal{K}$. We define the MD-strategy ζ from C by letting $\zeta((c, choice)) = \alpha$ if and only if $c \in C$. For any strategy $\mathfrak{T}$ for player 1, we can again view $\mathfrak{T}$ also as a scheduler for $\mathcal{K}$. Equation (5.6) holds again and shows that the expected accumulated weight in $\mathcal{G}$ is positive if player 0 plays according to ζ against any strategy for player 1. This finishes the proof of Claim 3.

Putting together Claims 1-3. We conclude that there is an SPR cause C in the original MDP $\mathcal{M}$ with $fscore(C) > \vartheta$ if and only if player 1 has a winning strategy in the constructed game $\mathcal{G}$. As both players have optimal MD-strategies in $\mathcal{G}$ [PB99], the decision problem is in NP$\cap$coNP: We can guess the MD-strategy for player 0 and solve the resulting stochastic shortest path problem in polynomial time [BT91] to obtain an NP-upper bound. Likewise, we can guess the MD-strategy for player 1 and solve the resulting stochastic shortest path problem to obtain the coNP-upper bound. $\square$

Optimality and threshold constraints for GPR causes. Computing optimal GPR causes for either quality measure can be done in polynomial space by considering all cause candidates, checking (G) in coNP and computing the corresponding quality measure in polynomial time (Section 5.2). As the space can be reused after each computation, this results in polynomial space. However, we show that no polynomial-time algorithms can be expected as the corresponding threshold problems are NP-hard. Let GPR-covratio (resp. GPR-recall, GPR-f-score) denote the decision problems: Given $\mathcal{M}, \mathsf{Eff}$ and $\vartheta \in \mathbb{Q}$, decide whether there exists a GPR cause with coverage ratio (resp. recall, f-score) at least ϑ.

Theorem 5.16. *The problems GPR-covratio, GPR-recall and GPR-f-score are* NP-*hard and belong to* Σ^P_2. *For Markov chains, all three problems are* NP-*complete.* NP-*hardness even holds for tree-like Markov chains.*

Proof. Σ^P_2*-membership.* The algorithms for GPR-covratio, GPR-recall and GPR-f-score rely on the guess-and-check principle: they start by non-deterministically guessing a set $\mathsf{Cause} \subseteq S$, then check in coNP whether Cause constitutes a GPR cause (see Section 4) and finally check $recall(\mathsf{Cause}) \leqslant \vartheta$ (with standard techniques), resp. $covrat(\mathsf{Cause}) \leqslant \vartheta$, resp. $fscore(\mathsf{Cause}) \leqslant \vartheta$ (Theorem 5.5) in polynomial time. The alternation between the existential quantification for guessing Cause and the universal quantification for the coNP check of the GPR condition results in the complexity Σ^P_2 of the polynomial-time hierarchy.

NP-*membership for Markov chains.* NP-membership for all three problems within Markov chains is straightforward as we may non-deterministically guess a cause and check in polynomial time whether it constitutes a GPR cause and satisfies the threshold condition for the recall, coverage ratio or f-score.

NP-*hardness of GPR-recall and GPR-covratio.* With arguments as in the proof of Lemma 5.10, the problems GPR-recall and GPR-covratio are polynomially interreducible for Markov chains. Thus, it suffices to prove NP-hardness of GPR-recall. For this, we provide a polynomial reduction from the knapsack problem. The input of the latter are sequences $A_1, \ldots, A_n, A$ and $B_1, \ldots, B_n, B$ of positive natural numbers and the task is to decide whether there exists a subset I of $\{1, \ldots, n\}$ such that

$$\sum_{i \in I} A_i < A \qquad \text{and} \qquad \sum_{i \in I} B_i \geqslant B \tag{5.7}$$

Let K be the maximum of $A, A_1, \ldots, A_n, B, B_1, \ldots, B_n$ and $N = 8(n+1) \cdot (K+1)$. We then define

$$a_i = \frac{A_i}{N}, \qquad a = \frac{A}{N}, \qquad b_i = \frac{B_i}{N}, \qquad b = \frac{B}{N}.$$

Then, $a, a_1, \ldots, a_n, b, b_1, \ldots, b_n$ are positive rational numbers strictly smaller than $\frac{1}{8(n+1)}$, and (5.7) can be rewritten as:

$$\sum_{i \in I} a_i < a \qquad \text{and} \qquad \sum_{i \in I} b_i \geqslant b \tag{5.8}$$

For $i \in \{1, \ldots, n\}$, let $p_i = 2(a_i + b_i)$ and $w_i = \frac{b_i}{p_i} = \frac{1}{2} \cdot \frac{b_i}{a_i + b_i}$. Then, $0 < p_i < \frac{1}{2(n+1)}$ and $0 < w_i < \frac{1}{2}$. Moreover, $p_i \left(\frac{1}{2} - w_i \right) = a_i$ and $p_i \cdot w_i = b_i$. Hence, (5.8) can be rewritten as:

$$\sum_{i \in I} p_i \left(\frac{1}{2} - w_i \right) < a \qquad \text{and} \qquad \sum_{i \in I} p_i w_i \geqslant b$$

which again is equivalent to:

$$\frac{\sum_{i \in I_0} p_i w_i}{\sum_{i \in I_0} p_i} > \frac{1}{2} \qquad \text{and} \qquad \sum_{i \in I_0} p_i w_i \geqslant p_0 + b \tag{5.9}$$

where $p_0 = 2a$, $w_0 = 1$ and $I_0 = I \cup \{0\}$. Note that $a < \frac{1}{8(n+1)}$ and hence $p_0 < \frac{1}{4(n+1)}$.

Define a tree-shape Markov chain $\mathcal{M}$ with non-terminal states init, $s_0, s_1, \ldots, s_n$, and terminal states $\text{eff}_0, \ldots, \text{eff}_n, \text{eff}_{unc}$ and $\text{noeff}, \text{noeff}_1, \ldots, \text{noeff}_n$. Transition probabilities are as follows:

- $P(\text{init}, s_i) = p_i$ for $i = 0, \ldots, n$
- $P(\text{init}, \text{eff}_{unc}) = \frac{1}{2} - \sum_{i=0}^{n} p_i w_i$
- $P(\text{init}, \text{noeff}) = 1 - \sum_{i=0}^{n} p_i - P(\text{init}, \text{eff}_{unc})$,
- $P(s_i, \text{eff}_i) = w_i$, $P(s_i, \text{noeff}_i) = 1 - w_i$ for $i = 1, \ldots, n$
- $P(s_0, \text{eff}_0) = 1 = w_0$.

Note that $p_0 + p_1 + \ldots + p_n < \frac{1}{2}$ as all p_i's are strictly smaller than $\frac{1}{2(n+1)}$. As the w_i's are bounded by 1, this yields $0 < P(\text{init}, \text{eff}_{unc}) < \frac{1}{2}$ and $0 < P(\text{init}, \text{noeff}) < 1$.

The graph structure of $\mathcal{M}$ is indeed a tree and $\mathcal{M}$ can be constructed from the values $A, A_1, \ldots, A_n$ and $B, B_1, \ldots, B_n$ in polynomial time. Moreover, for $\mathsf{Eff} = \{\mathsf{eff}_{\mathsf{unc}}\} \cup \{\mathsf{eff}_i : i = 0, 1, \ldots, n\}$ we have:

$$\mathrm{Pr}_{\mathcal{M}}(\Diamond \mathsf{Eff}) \;\; = \;\; \sum_{i=0}^{n} p_i w_i + \mathsf{P}(\mathsf{init}, \mathsf{eff}_{\mathsf{unc}}) \;\; = \;\; \frac{1}{2}.$$

As the values $w_1, \ldots, w_n$ are strictly smaller than $\frac{1}{2}$, we have $\mathrm{Pr}_{\mathcal{M}}(\Diamond \mathsf{Eff} \mid \Diamond C) < \frac{1}{2}$ for each nonempty subset C of $\{s_1, \ldots, s_n\}$. Thus, the only candidates for GPR causes are the sets $C_I = \{s_i : i \in I_0\}$ where $I \subseteq \{1, \ldots, n\}$ where as before $I_0 = I \cup \{0\}$. Note that for all states $s \in C_I$ there is a path satisfying $(\neg C_I) \, \mathsf{U} \, s$. Thus, C_I is a GPR cause iff C_I satisfies (G). We have:

$$\mathrm{Pr}_{\mathcal{M}}(\Diamond \mathsf{Eff} \mid \Diamond C_I) \;\; = \;\; \frac{\displaystyle\sum_{i \in I_0} p_i w_i}{\displaystyle\sum_{i \in I_0} p_i}$$

and

$$\mathit{recall}(C_I) \;\; = \;\; \mathrm{Pr}_{\mathcal{M}}(\Diamond(C_I \wedge \Diamond \mathsf{Eff}) \mid \Diamond \mathsf{Eff}) \;\; = \;\; 2 \cdot \sum_{i \in I_0} p_i w_i.$$

Thus, C_I is a GPR cause with recall at least $2(p_0 + b)$ if and only if the two conditions in (5.9) hold, which again is equivalent to the satisfaction of the conditions in (5.7). But this yields that $\mathcal{M}$ has a GPR cause with recall at least $2(p_0 + b)$ if and only if the knapsack problem is solvable for the input $A, A_1, \ldots, A_n, B, B_1, \ldots, B_n$.

NP-*hardness of GPR-f-score.* Using similar ideas, we also provide a polynomial reduction from the knapsack problem. Let $A, A_1, \ldots, A_n, B, B_1, \ldots, B_n$ be an input for the knapsack problem. We replace the A-sequence with $a, a_1, \ldots, a_n$ where $a = \frac{A}{N}$ and $a_i = \frac{A_i}{N}$ where N is as before. The topological structure of the Markov chain that we are going to construct is the same as in the NP-hardness proof for GPR-recall.

We will define polynomial-time computable values $p_0, p_1, \ldots, p_n \in {]0, 1[}$ (where $p_i = \mathsf{P}(\mathsf{init}, s_i)$), $w_1, \ldots, w_n \in {]0, 1[}$ (where $w_i = \mathsf{P}(s_i, \mathsf{eff}_i)$) and auxiliary variables $\delta \in {]0, 1[}$ and $\lambda > 1$ such that:

(1) $p_0 + p_1 + \ldots + p_n < \frac{1}{2}$

(2) $\lambda = \frac{p_0 + \frac{1}{2} - \delta}{p_0}$

(3) for all $i \in \{1, \ldots, n\}$:

    (3.1) $a_i = p_i\left(\frac{1}{2} - w_i\right)$ (in particular $w_i < \frac{1}{2}$)

    (3.2) $B_i = \frac{1}{\delta} B p_i(\lambda w_i - 1)$ (in particular $w_i > \frac{1}{\lambda}$)

Assuming such values have been defined, we obtain:

$$\sum_{i \in I} B_i \;\geqslant\; B \quad \text{iff} \quad \frac{1}{\delta} B \sum_{i \in I} p_i(\lambda w_i - 1) \;\geqslant\; B$$

$$\text{iff} \quad \sum_{i \in I} p_i(\lambda w_i - 1) \;\geqslant\; \delta$$

$$\text{iff} \quad \lambda \sum_{i \in I} p_i w_i \;\geqslant\; \delta + \sum_{i \in I} p_i$$

Hence:

$$\sum_{i \in I} B_i \geqslant B \quad \text{iff} \quad \frac{\displaystyle\sum_{i \in I} p_i w_i}{\delta + \displaystyle\sum_{i \in I} p_i} \geqslant \frac{1}{\lambda}$$

For all positive real numbers x, y, u, v with $\frac{x}{y} = \frac{1}{\lambda}$ we have:

$$\frac{x+u}{y+v} \geqslant \frac{1}{\lambda} \quad \text{iff} \quad \frac{u}{v} \geqslant \frac{1}{\lambda}$$

By the constraints for λ (see (2)), we have $\frac{p_0}{p_0 + \frac{1}{2} - \delta} = \frac{1}{\lambda}$. Therefore:

$$\frac{\displaystyle\sum_{i \in I} p_i w_i}{\delta + \displaystyle\sum_{i \in I} p_i} \geqslant \frac{1}{\lambda} \quad \text{iff} \quad \frac{p_0 + \displaystyle\sum_{i \in I} p_i w_i}{(p_0 + \frac{1}{2} - \delta) + \delta + \displaystyle\sum_{i \in I} p_i} = \frac{p_0 + \displaystyle\sum_{i \in I} p_i w_i}{p_0 + \frac{1}{2} + \displaystyle\sum_{i \in I} p_i} \geqslant \frac{1}{\lambda}$$

As before let $w_0 = 1$ and $I_0 = I \cup \{0\}$. Then, the above yields:

$$\sum_{i \in I} B_i \geqslant B \quad \text{iff} \quad \frac{\displaystyle\sum_{i \in I_0} p_i w_i}{\frac{1}{2} + \displaystyle\sum_{i \in I_0} p_i} \geqslant \frac{1}{\lambda}$$

As in the NP-hardness proof for GPR-recall and using (3.1):

$$\mathrm{Pr}_{\mathcal{M}}(\lozenge \mathsf{Eff}) = \frac{1}{2} > w_i \quad \text{for } i = 1, \dots, n$$

Thus, each GPR cause must have the form $C_I = \{s_i : i \in I_0\}$ for some subset I of $\{1, \dots, n\}$. Moreover:

$$\mathrm{Pr}_{\mathcal{M}}(\lozenge C_I) = \sum_{i \in I_0} p_i \quad \text{and} \quad \mathrm{Pr}_{\mathcal{M}}(\lozenge(C_I \wedge \lozenge \mathsf{Eff})) = \sum_{i \in I_0} p_i w_i$$

So, the f-score of C_I is:

$$fscore(C_I) = 2 \cdot \frac{\mathrm{Pr}_{\mathcal{M}}(\lozenge(C_I \wedge \lozenge \mathsf{Eff}))}{\mathrm{Pr}_{\mathcal{M}}(\lozenge \mathsf{Eff}) + \mathrm{Pr}_{\mathcal{M}}(\lozenge C_I)} = 2 \cdot \frac{\displaystyle\sum_{i \in I_0} p_i w_i}{\frac{1}{2} + \displaystyle\sum_{i \in I_0} p_i}$$

This implies:

$$\sum_{i \in I} B_i \geqslant B \quad \text{iff} \quad fscore(C_I) \geqslant \frac{2}{\lambda}$$

With $p_0 = 2a$ and using (3.1) and arguments as in the NP-hardness proof for GPR-recall, we obtain:

$$\sum_{i \in I} A_i < A \quad \text{iff} \quad C_I \text{ is a GPR cause}$$

Thus, the constructed Markov chain has a GPR cause with f-score at least $\frac{2}{\lambda}$ if and only if the knapsack problem is solvable for the input $A, A_1, \dots, A_n, B, B_1, \dots, B_n$.

It remains to define the values $p_1, \dots, p_n, w_1, \dots, w_n$ and δ. (The value of λ is then obtained by (2).) (3.1) and (3.2) can be rephrased as equations for w_i:

(3.1'): $w_i = \frac{1}{2} - \frac{a_i}{p_i}$

(3.2'): $w_i = \frac{1}{\lambda}\left(\delta\frac{B_i}{Bp_i} + 1\right)$

This yields an equation for p_i:

$$\frac{1}{2} - \frac{a_i}{p_i} = \frac{1}{\lambda}\left(\delta\frac{B_i}{Bp_i} + 1\right)$$

and leads to:

$$p_i = \frac{2\lambda}{\lambda-2}a_i + \frac{2\delta}{\lambda-2}\cdot\frac{B_i}{B} \tag{5.10}$$

We now substitute λ by (2) and arrive at

$$p_i = \frac{p_0}{\frac{1}{2}-\delta}a_i + a_i + \frac{\delta p_0}{\frac{1}{2}-\delta}\frac{B_i}{B}.$$

By choice of N, all a_i's and a are smaller than $\frac{1}{8(n+1)}$. Using this together with $p_0 = 2a$, we get:

$$p_i < \frac{1}{4(n+1)(\frac{1}{2}-\delta)}\frac{1}{8(n+1)} + \frac{1}{8(n+1)} + \frac{\delta}{4(n+1)(\frac{1}{2}-\delta)}\frac{B_i}{B} \tag{5.11}$$

Let now $\delta = \frac{1}{8K}$ (where K is as above, i.e., the maximum of the values $A, A_1, \ldots, A_n, B, B_1, \ldots, B_n$). Then, $p_1, \ldots, p_n$ are computable in polynomial time, and so are the values $w_1, \ldots, w_n$ (by (3.1')). As $\frac{2\lambda}{\lambda-2} > 2$ and using (5.10), we obtain $p_i > 2a_i$. So, by (3.1') we get $0 < w_i < \frac{1}{2}$.

It remains to prove (1). Using $\delta = \frac{1}{8K}$, we obtain from (5.11):

$$p_i < \frac{1}{4(n+1)(\frac{1}{2}-\frac{1}{8K})}\frac{1}{8(n+1)} + \frac{1}{8(n+1)} + \frac{1}{32(n+1)(\frac{1}{2}-\frac{1}{8K})K}\frac{B_i}{B} \overset{\text{def}}{=} x$$

As $\frac{1}{2} - \frac{1}{8K} \geqslant \frac{1}{4}$ and $\frac{B_i}{B} < K$, this yields:

$$p_i < x < \frac{1}{8(n+1)^2} + \frac{1}{8(n+1)} + \frac{1}{8(n+1)} < \frac{1}{2(n+1)}.$$

But then condition (1) holds. $\qquad\square$

Arbitrary quality measures. Consider any algebraic function $f(tp, tn, fp, fn)$. That is f satisfies some polynomial equation where the coefficients are polynomials in tp, tn, fp and fn. Almost every quality measure for binary classifiers (see [Pow11]) is such a function. Taking the worst case scheduler for such a function we define

$$f(\mathsf{Cause}) = \inf_{\mathfrak{S}} f^{\mathfrak{S}}(tp^{\mathfrak{S}}_{\mathsf{Cause}}, tn^{\mathfrak{S}}_{\mathsf{Cause}}, fp^{\mathfrak{S}}_{\mathsf{Cause}}, fn^{\mathfrak{S}}_{\mathsf{Cause}}),$$

where $\mathfrak{S}$ ranges over all schedulers such that $f^{\mathfrak{S}}$ is well defined. Given a PR cause Cause and a rational $\vartheta \in \mathbb{Q}$, deciding whether $f(\mathsf{Cause}) \leqslant \vartheta$ can be done in PSPACE as a satisfiability problem in the existential theory of the reals [Can88].

As we can decide for a given cause candidate Cause whether it is a SPR cause in P or a GPR cause in coNP, this also yields an algorithm for finding optimal causes for f. Given an MDP $\mathcal{M}$ with terminal effect set Eff and quality measure f as an algebraic function we consider each cause candidate Cause, check whether it is a PR cause (SPR or GPR) and consider the decision problem $f(\mathsf{Cause}) \leqslant \vartheta$. As all of these steps have a complexity upper bound of PSPACE and we only need to save the best cause candidate so far with its value $f(\mathsf{Cause})$, this results in polynomial space as well.

6. ω-REGULAR EFFECT SCENARIOS

In this section, we turn to an extension of the previous definition of PR causes. So far, we considered both the cause and the effect as sets of states in an MDP $\mathcal{M}$ with state space S. We will refer to this setting as the *state-based* setting from now on. In a more general approach, we now consider the effect to be an ω-regular language $rEff \subseteq S^\omega$ over the state space S. Note that we denote regular events as effects mainly by rEff to avoid confusion with effects as sets of states.

In a first step, we still consider sets of states Cause $\subseteq$ S as causes, which we call *reachability causes* (Section 6.1). For reachability GPR causes, the techniques from the previous section are mostly still applicable. For reachability SPR causes, on the other hand, we observe that they take on the flavor of state-based GPR causes as well. Afterwards, we generalize the definition further to allow ω-regular co-safety properties over the state space S as causes, which we call *co-safety causes* (Section 6.2). While this allows us to express much more involved cause-effect relationships, we will see that attempts of checking co-safety SPR causality or of finding good causes for a given effect encounter major new difficulties.

6.1. **Sets of states as causes.** Throughout this section, let $\mathcal{M} = (S, Act, P, init)$ be an MDP. As long as we use sets of states as causes, the definition of GPR and SPR causes can easily be adapted to ω-regular effects:

Definition 6.1 (Reachability GPR/SPR causes). Let $\mathcal{M}$ be as above. Let $rEff \subseteq S^\omega$ be an ω-regular language over S and Cause a nonempty subset of S such that for each $c \in$ Cause, there is a scheduler $\mathfrak{S}$ with $Pr_{\mathcal{M}}^{\mathfrak{S}}((\neg Cause) \cup c) > 0$. Then, Cause is said to be a *reachability GPR cause* for rEff iff the following condition (rG) holds:

(rG): For each scheduler $\mathfrak{S}$ where $Pr_{\mathcal{M}}^{\mathfrak{S}}(\Diamond Cause) > 0$:

$$Pr_{\mathcal{M}}^{\mathfrak{S}}(rEff \mid \Diamond Cause) \; > \; Pr_{\mathcal{M}}^{\mathfrak{S}}(rEff). \tag{rGPR}$$

Cause is called a *reachability SPR cause* for rEff iff the following condition (rS) holds:

(rS): For each state $c \in$ Cause and each scheduler $\mathfrak{S}$ where $Pr_{\mathcal{M}}^{\mathfrak{S}}((\neg Cause) \cup c) > 0$:

$$Pr_{\mathcal{M}}^{\mathfrak{S}}(rEff \mid (\neg Cause) \cup c) \; > \; Pr_{\mathcal{M}}^{\mathfrak{S}}(rEff). \tag{rSPR}$$

There is one small caveat that we want to mention here: If the effect rEff is a reachability property $\Diamond Eff$ for a set of states $Eff \subseteq S$, then this new definition allows for GPR/SPR causes Cause not disjoint from the set of states Eff. If two sets Cause, Eff $\subseteq$ S are disjoint, however, then Cause is a GPR/SPR cause for Eff according to Definition 3.1 iff Cause is a reachability GPR/SPR cause for the ω-regular event $\Diamond Eff$ according to the new definition. As we now view the effect as the ω-regular property on infinite executions, one can, nevertheless, argue that the *temporal priority* (C2) is captured by the new definition since the cause will be reached after finitely many steps if it is reached. We will address problems with this interpretation and a stronger notion of temporal priority in Section 6.1.3.

A first simple observation that follows as in the state-based setting is that a reachability SPR cause for rEff is also a reachability GPR cause for rEff

6.1.1. *Checking causality and existence of reachability PR causes.* To explore how this change of definition influences the previously established results for GPR and SPR causes, we have to clarify how effects will be represented. We use deterministic Rabin automata (DRAs) as they are expressive enough to capture all ω-regular languages and they are deterministic, which will allow us to form well-behaved products of the automata with MDPs. Let $\mathcal{M}$ be an MDP, rEff an effect given by the DRA $\mathcal{A}_{\mathsf{rEff}}$ and Cause $\subseteq S$ a cause candidate. As a special case we again have Markov chains with no non-deterministic choices. Then, the conditions (rG) and (rS) can easily be checked by computing the corresponding probabilities in polynomial time (see [BKKM14] for algorithms to compute conditional probabilities in MCs for path properties.) We now consider the case where non-deterministic choices exist. We will provide a model transformation of $\mathcal{M}$ using the DRA such that the resulting MDP has no end components and the effect is a reachability property again.

Notation 6.2 (Removing end components). Let $\mathcal{M}$ and $\mathcal{A}_{\mathsf{rEff}}$ be as above. Consider the product MDP $\mathcal{N} \stackrel{\text{def}}{=} \mathcal{M} \otimes \mathcal{A}_{\mathsf{rEff}}$. This product is an MDP equipped with a Rabin acceptance condition found in the second component of each state in the product.

We now take two copies of $\mathcal{N}$ representing a mode before Cause has been reached and one mode after Cause has been reached. So, each state $\mathfrak{s}$ is equipped with one extra bit 0 or 1. Initially, the MDP starts in the copy labeled with 0 and behaves like $\mathcal{N}$ until a state with its first component in Cause is reached. From there, the process moves to the corresponding successor states in the second copy labeled with 1. We call the resulting MDP $\mathcal{N}'$ and denote the set of states with their first component in Cause in the first copy that are reachable in $\mathcal{N}'$ by $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$, in particular, to express that these states are enriched with states of the automaton $\mathcal{A}_{\mathsf{rEff}}$.

Next, we consider the MECs $\mathcal{E}_1, \ldots, \mathcal{E}_k$ of $\mathcal{N}'$. Note that the states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ are not contained in any MEC. Furthermore, all MECs consist either only of states from the first copy labeled 0, or only of states from the second copy labeled with 1. For each MEC $\mathcal{E}_i$, we determine whether there is a scheduler for $\mathcal{E}_i$ that ensures the event $Acc(\mathcal{A}_{\mathsf{rEff}})$ that the acceptance condition of $\mathcal{A}_{\mathsf{rEff}}$ is met with probability 1 and whether there is a scheduler that ensures this event with probability 0. With the techniques of [dA97] and [BGC09] this can be done in polynomial time. We then add four new terminal states $\mathsf{eff}_{\mathsf{cov}}$, $\mathsf{noeff}_{\mathsf{fp}}$, $\mathsf{eff}_{\mathsf{unc}}$, and $\mathsf{noeff}_{\mathsf{tn}}$ and construct the MEC-quotient of $\mathcal{N}'$ while, for each $i \leqslant k$, enabling a new action in the state $s_{\mathcal{E}_i}$ obtained from $\mathcal{E}_i$ leading to

- $\mathsf{eff}_{\mathsf{unc}}$ if $Acc(\mathcal{A}_{\mathsf{rEff}})$ can be ensured with probability 1 in $\mathcal{E}_i$ and $\mathcal{E}_i$ is contained in copy 0, and
- another new action leading to $\mathsf{noeff}_{\mathsf{tn}}$ if $Acc(\mathcal{A}_{\mathsf{rEff}})$ can be ensured with probability 0 in $\mathcal{E}_i$ and $\mathcal{E}_i$ is contained in copy 0;
- $\mathsf{eff}_{\mathsf{cov}}$ if $Acc(\mathcal{A}_{\mathsf{rEff}})$ can be ensured with probability 1 in $\mathcal{E}_i$ and $\mathcal{E}_i$ is contained in copy 1, and
- another new action leading to $\mathsf{noeff}_{\mathsf{fp}}$ if $Acc(\mathcal{A}_{\mathsf{rEff}})$ can be ensured with probability 0 in $\mathcal{E}_i$ and $\mathcal{E}_i$ is contained in copy 1.

Finally, we remove all states which are not reachable (from the initial state). We call the resulting MDP $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ and emphasize that this MDP contains all states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$, has no end components, and has the four terminal states $\mathsf{eff}_{\mathsf{cov}}$, $\mathsf{noeff}_{\mathsf{fp}}$, $\mathsf{eff}_{\mathsf{unc}}$, and $\mathsf{noeff}_{\mathsf{tn}}$. Furthermore, for each $c \in \mathsf{Cause}$, we denote the subset of states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ with c in their first component by $c_{\mathcal{A}_{\mathsf{rEff}}}$. ◁

Lemma 6.3. *Let $\mathcal{M}$, Cause, and $\mathcal{A}_{\mathsf{rEff}}$ be as above and let $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ be the constructed MDP in Notation 6.2 which contains the set $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$. Then, Cause is a reachability GPR cause for rEff in $\mathcal{M}$ if and only if $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ is a GPR cause for $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$. Furthermore, Cause is a reachability SPR cause for rEff in $\mathcal{M}$ if and only if, for each $c \in \mathsf{Cause}$, the set $c_{\mathcal{A}_{\mathsf{rEff}}}$ of states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ with c in their first component is a GPR cause for $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$.*

Proof. The set $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ satisfies $\mathrm{Pr}^{\max}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\neg\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}} \cup \mathsf{d}) > 0$ for each state $\mathsf{d} \in \mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ by construction, since all states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ are reachable and a run cannot reach two different states in $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$. So, this minimality condition is satisfied in any case and, of course, $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ and $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ are disjoint.

Now, let $\mathfrak{T}$ be a scheduler for $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$. The scheduler $\mathfrak{T}$ can be mimicked by a scheduler $\mathfrak{S}$ for $\mathcal{M}$: As long as $\mathfrak{T}$ moves through the MEC-quotient of $\mathcal{N}'$, the scheduler $\mathfrak{S}$ follows this behavior by leaving MECs through the corresponding actions in $\mathcal{M}$. Whenever $\mathfrak{T}$ moves to one of the states $\mathsf{eff}_{\mathsf{cov}}$ and $\mathsf{eff}_{\mathsf{unc}}$, the last step begins in a state $s_{\mathcal{E}}$ obtained from a MEC $\mathcal{E}$ of $\mathcal{M}$. In this case, $\mathfrak{S}$ will stay in $\mathcal{E}$ while ensuring with probability 1 that the resulting run is accepted by $\mathcal{A}_{\mathsf{rEff}}$. Similarly, if $\mathfrak{T}$ moves to $\mathsf{noeff}_{\mathsf{tn}}$ or $\mathsf{noeff}_{\mathsf{fp}}$, the scheduler $\mathfrak{S}$ for $\mathcal{M}$ stays in the corresponding MEC and realizes the acceptance condition of $\mathcal{A}_{\mathsf{rEff}}$ with probability 0. Vice versa, a scheduler $\mathfrak{S}$ for $\mathcal{M}$ can be mimicked by a scheduler $\mathfrak{T}$ for $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ analogously. So in an end component $\mathcal{E}$, if $\mathfrak{S}$ stays in $\mathcal{E}$ and ensures that the resulting run is accepted by $\mathcal{A}_{\mathsf{rEff}}$ with probability p_1, stays in $\mathcal{E}$ and ensures that the resulting run is not accepted by $\mathcal{A}_{\mathsf{rEff}}$ with probability p_2, and leaves $\mathcal{E}$ with probability p_3, $\mathfrak{T}$ will move to the corresponding state $\mathsf{eff}_{\mathsf{cov}}$ or $\mathsf{eff}_{\mathsf{unc}}$ with probability p_1, to $\mathsf{noeff}_{\mathsf{tn}}$ or $\mathsf{noeff}_{\mathsf{fp}}$ with probability p_2, and leave take actions leaving $s_{\mathcal{E}}$ to other states with the same probability distribution with which $\mathfrak{S}$ takes the leaving actions of $\mathcal{E}$.

For such a pair of schedulers $\mathfrak{S}$ and $\mathfrak{T}$, we observe that

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond\mathsf{Cause}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}), \tag{6.1}$$

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond E), \text{ and} \tag{6.2}$$

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond\mathsf{Cause}\wedge\mathsf{rEff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond\mathsf{eff}_{\mathsf{cov}}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}\wedge\Diamond E), \tag{6.3}$$

where $E = \{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$. Hence, $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ is a GPR cause for $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ if and only if Cause is a reachability GPR cause for rEff in $\mathcal{M}$.

Now consider each element $c \in \mathsf{Cause}$ individually. We can use the same argumentation to see that a scheduler $\mathfrak{T}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ can be mimicked by a scheduler $\mathfrak{S}$ for $\mathcal{M}$, and vice versa, such that

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\neg\mathsf{Cause}\cup c) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond c_{\mathcal{A}_{\mathsf{rEff}}}), \text{ and}$$

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff} \mid \neg\mathsf{Cause}\cup c) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}}(\Diamond\mathsf{eff}_{\mathsf{cov}} \mid \Diamond c_{\mathcal{A}_{\mathsf{rEff}}}).$$

Thus, Cause is a reachability SPR cause for rEff in $\mathcal{M}$ if and only if $c_{\mathcal{A}_{\mathsf{rEff}}}$ is a GPR cause for $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ for all $c \in \mathsf{Cause}$. $\square$

This Lemma 6.3 shows that reachability SPR causality shares similarities with GPR causality. Our algorithmic results for reachability PR causes stem from the reduction provided by Notation 6.2 and Lemma 6.3. As an immediate consequence we can check conditions (rG) and (rS) in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ by using the already established algorithms for GPR causes. This results in the following complexity upper bounds.

Corollary 6.4. *Let $\mathcal{M}$ be an MDP and $\mathsf{rEff} \subseteq S^{\omega}$ an ω-regular language given as DRA $\mathcal{A}_{\mathsf{rEff}}$. Given a set $\mathsf{Cause} \subseteq S$ we can decide whether Cause is a reachability SPR/GPR cause for rEff in* coNP.

Proof. By Lemma 6.3, we can use the construction of $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$, which takes polynomial time. Then, Theorem 4.11 allows us to check whether Cause is a reachability GPR cause in coNP directly, while we can apply the GPR check to each set $c_{\mathcal{A}_{\mathsf{rEff}}}$ and the effect $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ for $c \in \mathsf{Cause}$ in order to determine whether Cause is a reachability SPR Cause for rEff. $\square$

As in the state-based setting, we can argue that there is a reachability GPR cause iff there is a reachability SPR cause iff there is a singleton reachability SPR cause. Consequently, the existence of a reachability GPR/SPR cause can be checked by checking for each state c of the MDP whether $\{c\}$ constitutes a reachability SPR cause. We conclude:

Corollary 6.5. *The existence of a reachability GPR/SPR cause can be decided in* coNP.

6.1.2. *Computing quality measures of reachability PR causes.* As in Section 5.1, we can view reachability PR causes as binary classifiers. This leads to the confusion matrix as before (Figure 13) with the difference that this time the path does not "hit" the effect set, but rather rEff holds on the path. Hence, we define the following entries of the confusion matrix: Given rEff, Cause and a scheduler $\mathfrak{S}$, we let

$$\mathsf{tp}^{\mathfrak{S}} \stackrel{\text{def}}{=} \mathsf{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond\mathsf{Cause}\wedge\mathsf{rEff}), \qquad \mathsf{tn}^{\mathfrak{S}} \stackrel{\text{def}}{=} \mathsf{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\neg\Diamond\mathsf{Cause}\wedge\neg\mathsf{rEff}),$$

$$\mathsf{fp}^{\mathfrak{S}} \stackrel{\text{def}}{=} \mathsf{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond\mathsf{Cause}\wedge\neg\mathsf{rEff}), \quad \mathsf{fn}^{\mathfrak{S}} \stackrel{\text{def}}{=} \mathsf{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\neg\Diamond\mathsf{Cause}\wedge\mathsf{rEff}).$$

Lemma 6.6. *Let $\mathcal{M}$ be as above,* rEff *an ω-regular effect given by the DRA $\mathcal{A}_{\mathsf{rEff}}$ and let* Cause $\subseteq S$ *be a reachability GPR/SPR cause. Further, let $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ be as constructed above with the set of cause states* $\mathsf{Cause}_{\mathcal{A}_{\mathsf{rEff}}}$ *and the set of effect states* $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$. *Then, for each scheduler $\mathfrak{S}$ for $\mathcal{M}$, there is a scheduler $\mathfrak{T}$ for $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$, and vice versa, such that*

$$\mathsf{tp}^{\mathfrak{S}} = \mathsf{tp}^{\mathfrak{T}}, \qquad \mathsf{tn}^{\mathfrak{S}} = \mathsf{tn}^{\mathfrak{T}}, \qquad \mathsf{fp}^{\mathfrak{S}} = \mathsf{fp}^{\mathfrak{T}}, \qquad \mathsf{fn}^{\mathfrak{S}} = \mathsf{fn}^{\mathfrak{T}},$$

where the values for $\mathfrak{T}$ are defined in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ as for the state-based setting (cf. Section 5.1).

Proof. In the proof of Lemma 6.3, we have seen that for each scheduler $\mathfrak{S}$ for $\mathcal{M}$, we can find a scheduler $\mathfrak{T}$ for $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ and vice versa such that Equations (6.1) - (6.3) hold. This implies the equalities claimed here. $\square$

Analogously to Section 5.1, we can now define *recall*, *covrat*, and *fscore* of a reachability PR cause as the infimum of these values in terms of $\mathsf{tp}^{\mathfrak{S}}$, $\mathsf{fp}^{\mathfrak{S}}$, $\mathsf{tn}^{\mathfrak{S}}$, and $\mathsf{fn}^{\mathfrak{S}}$ over all schedulers $\mathfrak{S}$ for which the respective quality measures are defined. The computation of these values can then be done with the methods from the state-based setting:

Corollary 6.7. *Let $\mathcal{M}$ be an MDP and* rEff $\subseteq S^{\omega}$ *an ω-regular language given as DRA $\mathcal{A}_{\mathsf{rEff}}$. Given a reachability SPR/GPR cause* Cause $\subseteq S$ *we can compute* recall(Cause), covrat(Cause) *and* fscore(Cause) *in polynomial time.*

Proof. Use Lemma 6.6, Corollary 5.4 and Theorem 5.5. $\square$

6.1.3. *Finding quality optimal reachability PR causes.* When trying to find good causes for an ω-regular effect rEff, we cannot say that effects and causes should be disjoint as in the setting where effects and causes were sets of states. This leads to the possibility that causes might exist that do not capture the intuition behind temporal priority: E.g., if the effect rEff is a reachability property $\Diamond E$ for a set of states E, the set of states E itself will be a reachability PR cause if for each state $c \in E$, $\mathsf{Pr}^{\max}_{\mathcal{M}}(\neg E \cup c) > 0$ holds. Furthermore, there might be causes C that can only be reached after E has already been reached.

In order to account for the temporal priority of causes, we will include the following condition when trying to find good causes: We require for a cause $\mathsf{Cause} \subseteq \mathsf{S}$ that

$$\mathrm{Pr}_{\mathcal{M}}^{\min}(\mathsf{rEff} \mid \neg\mathsf{Cause}\,\mathsf{U}\,c) < 1 \text{ for all } c \in \mathsf{Cause}. \qquad\qquad \text{(TempPrio)}$$

Intuitively, this states that it is never already certain that the execution will belong to rEff when the cause is reached. A variation of this criterium has been proposed also in [BDF$^+$21].

Remark 6.8. The condition (TempPrio) could be added to the definition of reachability PR causes. After the product construction in Notation 6.2, the condition can easily be checked for a given cause candidate $\mathsf{Cause} \subseteq \mathsf{S}$: For each $c \in \mathsf{Cause}$, there must be at least one state with c in the first component in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ such that $\mathsf{noeff}_{\mathsf{fp}}$ is reachable from this state.

Furthermore, this condition is stronger than the requirement that causes and effects are disjoint in the state-based setting. In the state-based setting, however, the analogue of condition (TempPrio) could also be included easily. Instead of having to be disjoint from a set of effect states Eff, a cause Cause would then simply not be allowed to contain any state s with $\mathrm{Pr}_{\mathcal{M},s}^{\min}(\Diamond\mathsf{Eff}) = 1$. $\qquad\qquad \triangleleft$

Now, we want to find recall- and coverage ratio-optimal reachability SPR causes. As in the state-based setting, we define the set $\mathcal{C}$ of all possible singleton reachability SPR causes for rEff that in addition satisfy (TempPrio) as explained in Remark 6.8. By Corollary 6.4, we can check whether a state $c \in \mathsf{S}$ is a singleton reachability SPR cause in coNP; whether there is at least one state with c in the first component in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$ such that $\mathsf{noeff}_{\mathsf{fp}}$ is reachable from this state is checkable in polynomial time. Thus, we can again define the set of singleton reachability SPR causes $\mathcal{C} = \{s \in \mathsf{S} \mid s \text{ is reachability SPR cause satisfying } \mathrm{Pr}_{\mathcal{M}}^{\min}(\mathsf{rEff} \mid \neg\mathsf{Cause}\,\mathsf{U}\,c) < 1\}$. As before, the canonical cause $\mathsf{CanCause}$ is now the set of states $c \in \mathcal{C}$ for which there is a scheduler $\mathfrak{S}$ with $\mathrm{Pr}^{\mathfrak{S}}(\neg\mathcal{C}\,\mathsf{U}\,c)$.

For the complexity of the computation of the recall- and coverage ratio-optimal canonical cause and its values, the observations above lead us to the complexity class $\mathrm{PF}^{\mathrm{NP}}$ as defined in [Sel94]. It consists of all functions that can be computed in polynomial time with access to an NP-oracle, or equivalently a coNP-oracle.

Proposition 6.9. *If $\mathcal{C} \neq \emptyset$ then* $\mathsf{CanCause}$ *is a ratio- and recall optimal reachability SPR cause for* rEff. *The threshold problem for the coverage ratio and the recall can be decided in* coNP. *The optimal values recall*($\mathsf{CanCause}$) *and covrat*($\mathsf{CanCause}$) *can be computed in* $\mathrm{PF}^{\mathrm{NP}}$.

Proof. The optimality of $\mathsf{CanCause}$ follows by the arguments used for Theorem 5.11. In order to compute $\mathsf{CanCause}$ we check for each state s whether (rS) does not hold in NP and then take the remaining states $c \in \mathsf{S}$ to define $\mathcal{C}$ by checking (TempPrio) for each c. If (TempPrio) holds then $c \in \mathcal{C}$ and $\mathsf{CanCause} = \{c \in \mathcal{C} \mid \mathrm{Pr}_{\mathcal{M}}^{\max}(\neg\mathcal{C}\,\mathsf{U}\,c) > 0\}$. This allows to compute the recall- and coverage ratio-optimal cause $\mathsf{CanCause}$ in $\mathrm{PF}^{\mathrm{NP}}$.

For the threshold problem whether there is a reachability SPR cause with recall at least a given $\vartheta \in \mathbb{Q}$, the coNP upper bound can be shown as follows: For each state c that does not belong to $\mathcal{C}$, i.e., that is not a singleton reachability SPR cause, there is a polynomial size certificate for this, as it can be checked in coNP. The collection of all states that do not belong to $\mathcal{C}$ together with these certificates (they do not belong to $\mathcal{C}$) can now serve as a certificate that the optimal recall is less than ϑ in this the case. Given such a collection of certificates, we can check in polynomial time that indeed all provided states do not belong to $\mathcal{C}$. The complement of the provided states forms a super set $\mathcal{D}$ of $\mathsf{CanCause}$. Computing the recall of the set $\mathcal{D}$ can then be done in polynomial time as in Corollary 6.7. This value is an upper bound for the recall of $\mathsf{CanCause}$. Note that if all states that do not belong to $\mathcal{C}$ are given in the certificate, the value even equals the recall of $\mathsf{CanCause}$.

So, if the optimal value is less than ϑ, this is witnessed by the described certificate containing all states not in $\mathcal{C}$. Vice versa, if a certificate is given resulting in a super set $\mathcal{D}$ of CanCause such that the recall of $\mathcal{D}$ is less than ϑ, then there is no reachability SPR cause with a recall of at least ϑ. So, the threshold problem lies in coNP. For the coverage ratio, the analogous argument works. $\qquad\square$

For the threshold problems for f-score-optimal reachability SPR causes and reachability GPR causes optimal with respect to recall, coverage ration or f-score that satisfy (TempPrio), we rely on the guess-and-check approach used for optimal GPR causes in the state-based setting. We guess a subset Cause of states of the MDP, check whether we found a reachability SPR/GPR cause in coNP, and compute the quality measure under consideration in polynomial time as explained in the previous section. For the computation of an optimal cause, we obtain a polynomial-space algorithm.

Corollary 6.10. *Let $\mathcal{M}$ be an MDP and* rEff *be an ω-regular language given by $\mathcal{A}_{\mathsf{rEff}}$. Given $\vartheta \in \mathbb{Q}$, deciding whether there exists a reachability GPR cause Cause with recall(Cause) $\geqslant \vartheta$ (resp. covrat(Cause) $\geqslant \vartheta$, fscore(Cause) $\geqslant \vartheta$) can be done in Σ_2^P and is NP-hard. NP-hardness even holds for Markov chains.*

Deciding whether there is a reachability SPR cause Cause with fscore(Cause) $\geqslant \vartheta$ can be done in Σ_2^P. A recall-, covratio-, or f-score-optimal reachability GPR cause as well as an f-score-optimal reachability SPR cause can be computed in polynomial space.

Proof. Obviously the lower bounds extend to this setting as we can interpret GPR causes for Eff as reachability GPR causes for $\Diamond$Eff. The upper bounds extend to this setting by using the construction from Notation 6.2. Since for Theorem 5.16 we relied on guess-and-check algorithms to establish the upper bounds for the threshold problems, we can use analogous algorithms in setting of ω-regular effects. We guess a set Cause $\subseteq S$, check the reachability GPR causality in coNP (Corollary 6.4) and compute the value of the quality measure in polynomial time (Corollary 6.7). Again, the alternation between the existential quantification for guessing Cause and the universal quantification for the coNP check results in the complexity Σ_2^P of the polynomial-time hierarchy.

In order to show that the decision problem for *fscore*(Cause) $\geqslant \vartheta$ for SPR causes Cause is in Σ_2^P we resort to the constructed MDP $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$. By Lemma 6.3 a reachability SPR cause Cause in $\mathcal{M}$ corresponds to a set of GPR causes $\{c_{\mathcal{A}_{\mathsf{rEff}}} \mid c \in \mathsf{Cause}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$, which can be interpreted as GPR cause $C = \bigcup_{c \in \mathsf{Cause}} c_{\mathcal{A}_{\mathsf{rEff}}}$. This way we can encode the property *fscore*(Cause) $\geqslant \vartheta$ in $\mathcal{M}$ by *fscore*(C) $\geqslant \vartheta$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{Cause}]}$. This results in a decision problem GPR-f-score for GPR causes which is in Σ_2^P by Theorem 5.16.

For computing optimal GPR causes as well as f-score-optimal SPR causes we can try all cause candidates by computing the related value (*recall*, *covrat* or *fscore*) and always store the best one so far. As the space for the cause can be reused, this results in a polynomial space algorithm. $\qquad\square$

6.2. ω-regular co-safety properties as causes.

We now want to discuss an extension of the previous framework when we also consider causes to be regular sets of executions. However, in order to account for the *temporal priority* of causes, i.e., the fact that causes should occur before their effects, it makes sense to restrict causes to ω-regular co-safety properties. The reason is that an ω-regular co-safety property $\mathcal{L}$ is uniquely determined by the regular set of minimal *good* prefixes of words in $\mathcal{L}$. Recall that a good prefix π for $\mathcal{L}$ is a finite word such that all infinite extensions of π belong to $\mathcal{L}$ and that all infinite words in the co-safety language $\mathcal{L}$ have a good prefix. Hence, we can say that a cause rCause occurred as soon as a good prefix for rCause has been produced. For this subsection we will denote regular effects and causes mainly by rEff and rCause to avoid confusion with effects

and causes as sets of states. In the following formal definition, we use finite words $\sigma \in S^*$ to denote the event σS^ω.

Definition 6.11 (co-safety GPR/SPR causes)**.** Let $\mathcal{M}$ be an MDP with state space S and let $\mathrm{rEff} \subseteq S^\omega$ be an ω-regular language. An ω-regular co-safety language $\mathrm{rCause} \subseteq S^\omega$ is a *co-safety GPR cause* for rEff if the following condition (coG) holds:

(coG): For each scheduler $\mathfrak{S}$ where $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\mathrm{rCause}) > 0$:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\mathrm{rEff} \mid \mathrm{rCause}) \; > \; \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\mathrm{rEff}). \qquad\qquad \text{(cosafeGPR)}$$

The event rCause is called a *co-safety SPR cause* for rEff if the following condition (coS) hold:

(coS): For each minimal good prefix σ for rCause and each scheduler $\mathfrak{S}$ where $\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\sigma) > 0$:

$$\mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\mathrm{rEff} \mid \sigma) \; > \; \mathrm{Pr}_{\mathcal{M}}^{\mathfrak{S}}(\mathrm{rEff}). \qquad\qquad \text{(cosafeSPR)}$$

As in the state-based setting it follows that co-safety SPR cause are also co-safety GPR causes.

6.2.1. *Checking co-safety causality.* We will represent co-safety PR causes as DFAs which accept good prefixes of the represented ω-regular event. Note that, for any ω-regular co-safety property, there is a DFA accepting exactly the minimal good prefixes. So, we will restrict to such DFAs that accept the minimal good prefixes of an ω-regular co-safety property. Such a DFA can never accept a word w as well as a proper prefix v of w.

Let now $\mathcal{M}$ be an MDP, rEff an effect given by the DRA $\mathcal{A}_{\mathrm{rEff}}$ and rCause a cause candidate given by a DFA $\mathcal{A}_{\mathrm{rCause}}$ as above. So, in particular, $\mathcal{A}_{\mathrm{rCause}}$ accepts exactly the minimal good prefixes for rCause. We now want to check, whether rCause is a co-safety SPR cause (resp. co-safety GPR cause). For the special case of Markov chains the check can be done in polynomial time analogously to reachability PR causes by computing the corresponding conditional probabilities. We can provide a model transformation of $\mathcal{M}$ using both automata such that the resulting MDP has no end components and the effect is a reachability property again similar to Notation 6.2.

For this consider the product $\mathcal{N} \overset{\text{def}}{=} \mathcal{M} \otimes \mathcal{A}_{\mathrm{rEff}} \otimes \mathcal{A}_{\mathrm{rCause}}$. This product is an MDP equipped with two kinds of acceptance conditions. The Rabin acceptance of $\mathcal{A}_{\mathrm{rEff}}$ in the second component of each state and the acceptance condition of $\mathcal{A}_{\mathrm{rCause}}$ in the third component. Now let $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ be the set of all states of $\mathcal{N}$ whose third component is accepting in $\mathcal{A}_{\mathrm{rCause}}$ and which are reachable from the initial state.

As in Notation 6.2, we construct an MDP $\mathcal{N}'$ by introducing a mode before $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ and a mode after $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$. We then take the MEC-quotient with the four terminal states $\mathrm{eff}_{\mathrm{cov}}$, $\mathrm{noeff}_{\mathrm{fp}}$, $\mathrm{eff}_{\mathrm{unc}}$, and $\mathrm{noeff}_{\mathrm{tn}}$, which are reachable from states $s_{\mathcal{E}}$ that result from collapsing the MEC $\mathcal{E}$ depending on whether $\mathcal{E}$ is contained in the before- or after-$\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ mode and whether the acceptance condition of $\mathcal{A}_{\mathrm{rEff}}$ can be realized with probability 0 and 1, respectively in $\mathcal{E}$, analogously to Notation 6.2. We call the resulting MDP $\mathcal{M}_{[\mathrm{rEff},\mathrm{rCause}]}$ and emphasize that this MDP still contains all states in $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ as they are not contained in any end component.

We start with the observation, that for co-safety GPR causes this reduction characterizes the condition (coG) completely.

Lemma 6.12. *Let $\mathcal{M}$ be and MDP, $\mathcal{A}_{\mathrm{rEff}}$ an DRA, and $\mathcal{A}_{\mathrm{rCause}}$ a DFA be as above and let $\mathcal{M}_{[\mathrm{rEff},\mathrm{rCause}]}$ be the constructed MDP that contains the set $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ of reachable states that have an accepting $\mathcal{A}_{\mathrm{rCause}}$-component. Then, rCause is a co-safety GPR cause for rEff in $\mathcal{M}$ if and only if the set of states $\mathrm{rCause}_{\mathcal{A}_{\mathrm{rEff}}}$ is a GPR cause for $\{\mathrm{eff}_{\mathrm{cov}}, \mathrm{eff}_{\mathrm{unc}}\}$ in $\mathcal{M}_{[\mathrm{rEff},\mathrm{rCause}]}$.*

Proof. The set $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ satisfies $\mathrm{Pr}^{\max}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\neg \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \, \mathsf{U} \, c) > 0$ for each $c \in \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ by construction, since all states in $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ are reachable and a run cannot reach two different states in $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$. Thus, the minimality condition is satisfied in any case.

Now, let $\mathfrak{S}$ be a scheduler for $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$. The scheduler $\mathfrak{S}$ can be mimicked by a scheduler $\mathfrak{T}$ for $\mathcal{M}$: As long as $\mathfrak{S}$ moves through the MEC-quotient of $\mathcal{N}'$, the scheduler $\mathfrak{T}$ follows this behavior by leaving MECs through the corresponding actions in $\mathcal{M}$. Whenever $\mathfrak{S}$ moves to one of the states $\mathsf{eff}_{\mathsf{cov}}$ and $\mathsf{eff}_{\mathsf{unc}}$, the last step begins in a state $s_{\mathcal{E}}$ obtained from a MEC $\mathcal{E}$ of $\mathcal{M}$. In this case, $\mathfrak{T}$ will stay in $\mathcal{E}$ while ensuring with probability 1 that the resulting run is accepted by $\mathcal{A}_{\mathsf{rEff}}$. Similarly, if $\mathfrak{S}$ moves to $\mathsf{noeff}_{\mathsf{tn}}$ or $\mathsf{noeff}_{\mathsf{fp}}$, the scheduler $\mathfrak{T}$ for $\mathcal{M}$ stays in the corresponding MEC and realizes the acceptance condition of $\mathcal{A}_{\mathsf{rEff}}$ with probability 0. Vice versa, a scheduler $\mathfrak{T}$ for $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ can be mimicked by a scheduler $\mathfrak{S}$ for $\mathcal{M}$ analogously (see also the proof of Lemma 6.3).

For such a pair of schedulers $\mathfrak{S}$ and $\mathfrak{T}$, we observe that

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rCause}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}), \tag{6.4}$$

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{E}), \text{ and} \tag{6.5}$$

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rCause} \wedge \mathsf{rEff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{eff}_{\mathsf{cov}}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \wedge \Diamond \mathsf{E}). \tag{6.6}$$

Hence, $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ is a GPR cause for E in $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ if and only if rCause is a regular GPR cause for rEff in $\mathcal{M}$. $\square$

For co-safety SPR causes this is not a full characterization but only holds in one direction:

Lemma 6.13. *Let $\mathcal{M}$, $\mathcal{A}_{\mathsf{rEff}}$, and $\mathcal{A}_{\mathsf{rCause}}$ be as above and let $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ be the constructed MDP that contains the set $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ of reachable states that have an accepting $\mathcal{A}_{\mathsf{rCause}}$-component. If rCause is a co-safety SPR cause for rEff in $\mathcal{M}$, then the set of states $\mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ is an SPR cause for $\{\mathsf{eff}_{\mathsf{cov}}, \mathsf{eff}_{\mathsf{unc}}\}$ in $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$.*

Proof. Suppose rCause is a regular SPR cause for rEff in $\mathcal{M}$. Let $c \in \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}}$ and let $\mathfrak{T}$ be a scheduler for $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ such that $\mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\neg \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \, \mathsf{U} \, c) > 0$. Let

$$\Pi = \{\pi \text{ a path in } \mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]} \mid \pi \vDash \neg \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \, \mathsf{U} \, c \text{ and } \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\pi) > 0\}.$$

Let $\mathfrak{S}$ be a scheduler for $\mathcal{M}$ mimicking $\mathfrak{T}$ as described in the proof of Lemma 6.12. Let Σ_{π} be the set of $\mathfrak{T}$-paths in $\mathcal{M}$ that correspond to the path π in Π. I.e., a path σ belongs to Σ_{π} if it moves through the MECs of $\mathcal{M}$ in the same way as the path π moves through the MEC-quotient until σ and π reach the state c and if furthermore, σ has positive probability under $\mathfrak{S}$. Now,

$$\mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{E} \mid \neg \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \, \mathsf{U} \, c) = \frac{\sum_{\pi \in \Pi} \sum_{\sigma \in \Sigma_{\pi}} \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff} \mid \sigma) \cdot \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\sigma)}{\sum_{\pi \in \Pi} \sum_{\sigma \in \Sigma_{\pi}} \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\sigma)}.$$

All the terms $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff} \mid \sigma)$ are greater than $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\mathsf{rEff}) = \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{E})$. Hence, we conclude that

$$\mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{E} \mid \neg \mathsf{rCause}_{\mathcal{A}_{\mathsf{rEff}}} \, \mathsf{U} \, c) > \mathrm{Pr}^{\mathfrak{T}}_{\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}}(\Diamond \mathsf{E}). \qquad \square$$

For checking co-safety SPR causality this is not sufficient. The underlying problem, in which co-safety SPR causes and SPR causes differ, can be seen in the following example:

Example 6.14. Consider the MDP $\mathcal{M}$ from Figure 17 with $\mathsf{rEff} = \Diamond \mathsf{eff}$. For every scheduler $\mathfrak{S}$ with $\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond c) > 0$ we have

$$\mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond \mathsf{eff} \mid \Diamond c) > \mathrm{Pr}^{\mathfrak{S}}_{\mathcal{M}}(\Diamond \mathsf{eff})$$
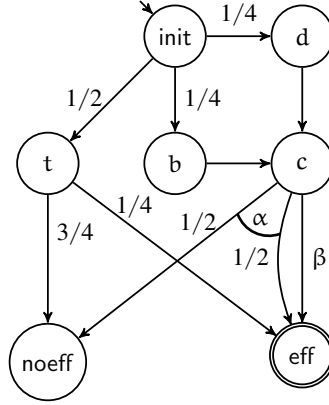
Figure 17: An MDP $\mathcal{M}$ for which the reduction from co-safety to state-based PR causes fails

and thus c is a state-based SPR cause for eff. On the other hand for the scheduler τ, which chooses α after the path $\pi = a\ b\ c$ and β otherwise ,we have

$$\mathrm{Pr}_{\mathcal{M}}^{\tau}(\lozenge \mathsf{eff} \mid \pi) = \frac{1}{2} = \mathrm{Pr}_{\mathcal{M}}^{\tau}(\lozenge \mathsf{eff}).$$

Therefore, the desired reduction as in Lemma 6.12 does not work for $\mathcal{M}$. Note that the violation of the condition (coS) is only possible in this example if the scheduler behaves differently depending on how state c is reached. This different behavior, however, does not have anything to do with the effect, and potentially different residual properties that have to be satisfied to achieve the effect; in the example, the effect is just a reachability property. Furthermore, we want to emphasize that the concrete probabilities of the individual paths leading to c are important for the violation. In general, this imposes a major challenge for checking the condition (coS), for which we do not know a solution. Similar problems arise when trying to check the existence of a co-safety SPR cause. A witness might be just one individual path, potentially only together with a scheduler that realizes this path with very low probability.   ◁

6.2.2. *Computation of quality measures of co-safety causes.* Analogously to Section 6.1.2, we can define recall, coverage ratio, and f-score of co-safety PR causes. With the construction of $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ and the correspondence between schedulers $\mathfrak{S}$ for $\mathcal{M}$ and $\mathfrak{T}$ for $\mathcal{M}_{[\mathsf{rEff},\mathsf{rCause}]}$ satisfying Equations (6.4)-(6.6) established in the proof of Lemma 6.12, we obtain analogously to the setting of reachability PR causes:

Corollary 6.15. *Let $\mathcal{M}$ be an MDP and $\mathsf{rEff} \subseteq S^{\omega}$ an ω-regular language given as DRA $\mathcal{A}_{\mathsf{rEff}}$. Given a co-safety SPR/GPR cause $\mathsf{rCause} \subseteq S^{\omega}$ by a DFA $\mathcal{A}_{\mathsf{rCause}}$, we can compute recall$(\mathsf{Cause})$, covrat$(\mathsf{Cause})$ and fscore(Cause) in polynomial time.*

6.2.3. *Finding optimal co-safety PR causes.* Already for reachability PR causes, we have seen that without further restrictions on the causes we allow, causes might be trivial and intuitively violate the idea of temporal priority (cf. Section 6.1.3). Hence, also here, we impose an additional condition, a variation of the condition (TempPrio) used above: In line with the difference between the definitions of reachability PR causes and co-safety PR causes, we require that after any good prefix σ of a
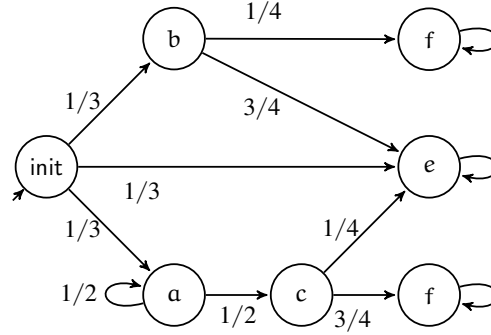
Figure 18: Markov chain $\mathcal{M}$ where the regular effect $\Diamond e$ has no recall-optimal regular PR-cause.

co-safety cause rCause, the probability that effect rEff will occur is not guaranteed to be 1, i.e., we require that

$$\mathrm{Pr}_{\mathcal{M}}^{\min}(\mathrm{rEff} \mid \sigma) < 1 \text{ for all good prefixes } \sigma \text{ of rCause.} \qquad \text{(TempPrio2)}$$

Unfortunately, we will observe that there are some obstacles in the way when trying to find optimal co-safety PR causes.

Following the observations from Theorem 5.11 we can define a canonical co-safety PR cause which is an optimal co-safety SPR cause for both recall and coverage ratio. In this fully path-based setting this canonical cause consists of all minimal paths which are singleton co-safety SPR causes. However, as we are not aware of a feasible way to check (coS), the computation of this cause is unclear.

For co-safety GPR causes, the following example illustrates that there might be no recall-optimal causes that respect (TempPrio2). Intuitively, the reason is that causes can be pushed arbitrarily close towards a violation of the probability raising condition while increasing the recall:

Example 6.16. This example will show that there is a Markov chain $\mathcal{M}$ with a state e such that the effect rEff $= \Diamond e$ has regular GPR causes that respect the condition (TempPrio2), but no recall-optimal co-safety GPR cause that respects (TempPrio2).

Consider the Markov chain $\mathcal{M}$ depicted in Figure 18 with states S and the effect rEff $= \Diamond e$. First of all, we have that $\mathrm{Pr}_{\mathcal{M}}(\mathrm{rEff}) = 2/3$. Furthermore, clearly the cause $\mathrm{init}\, b\, S^{\omega}$ with the unique minimal good prefix $\mathrm{init}\, b$ is a regular GPR cause for example as $\mathrm{Pr}_{\mathcal{M}}(\mathrm{rEff} \mid \mathrm{init}\, b) = 3/4$.

Next, note that there cannot be a regular GPR cause rCause$'$ that does not have $\mathrm{init}\, b$ as a minimal good prefix. By (TempPrio2) the minimal good prefixes are not allowed to end in e. Furthermore init is clearly also no candidate for a minimal good prefix of rCause$'$ as that would imply that rCause consists of all paths of $\mathcal{M}$ and cannot satisfy the condition (rGPR). If $\mathrm{init}\, b$ is not a minimal good prefix, hence all minimal good prefixes have to end in a, c, or f. Afterwards, the probability to reach e is at most $1/4$ and hence also the probability $\mathrm{Pr}_{\mathcal{M}}(\mathrm{rEff} \mid \mathrm{rCause}') \leqslant 1/4$ because it is a weighted average of the probabilities $\mathrm{Pr}_{\mathcal{M}}(\mathrm{rEff} \mid \sigma) \leqslant 1/4$ of the minimal good prefixes σ of rCause$'$.

So, all regular GPR causes have the minimal good prefix $\mathrm{init}\, b$ together with potentially further minimal good prefixes. Let now

$$\mathrm{rCause}_p \overset{\mathrm{def}}{=} \mathrm{init}\, b\, S^{\omega} \cup A_p$$

where A_p is a regular subset of the paths $\{\mathrm{init}\, a^k c \mid k \geqslant 1\}$ such that all the paths in A_p together have probability mass $\frac{1}{3}p$. Note that we can find such a set A_p for a dense set of values $p \in [0, 1]$.

We compute

$$\Pr_{\mathcal{M}}(\mathrm{rCause}_p) = \frac{1}{3}(1+p)$$

and

$$\Pr_{\mathcal{M}}(\mathrm{rCause}_p \wedge \mathrm{rEff}) = \frac{1}{4} + \frac{1}{12}p.$$

So, we obtain that rCause_p is a regular GPR cause if

$$\Pr_{\mathcal{M}}(\mathrm{rEff} \mid \mathrm{rCause}_p) = \frac{\frac{1}{4} + \frac{1}{12}p}{\frac{1}{3}(1+p)} > \frac{2}{3} = \Pr_{\mathcal{M}}(\mathrm{rEff}).$$

After multiplying the inequality with $12(1+p)$, we see that this holds iff $9 + 3p > 8 + 8p$ iff $p < \frac{1}{5}$. The recall of rCause_p is now

$$\Pr_{\mathcal{M}}(\mathrm{rCause}_p \mid \mathrm{rEff}) = \frac{\frac{1}{4} + \frac{1}{12}p}{2/3} = \frac{3}{8} + \frac{1}{8}p.$$

So, among the co-safety GPR causes of the form rCause_p, there is no recall-optimal one. For p tending to $1/5$ from below, the recall always increases. Note also that an ε-recall-optimal co-safety GPR cause for $\varepsilon > 0$ must take a very complicated form. It has to select paths of the form $\mathrm{init}\, a^k c$ that have probability $\frac{1}{3 \cdot 2^k}$ such that there probability adds up to a value less than, but close to $\frac{1}{15}$. ◁

We have seen that for recall-optimal (and hence coverage ratio-optimal) SPR causes for an effect given by $\mathcal{A}_{\mathrm{rEff}}$, we can provide a characterization of the canonical cause. How to compute this cause, however, is unclear as we do not know how to check the co-safety SPR condition and as there might be some paths ending in a given state in the product of $\mathcal{M}$ and $\mathcal{A}_{\mathrm{rEff}}$ that belong to the canonical cause while other paths ending in that state do not. For co-safety GPR causes, we have even seen that there might be no (non-trivial) optimal causes and that causes close to the optimum can be required to take a very complicated shape. As the f-score is a more involved quality measure than the recall, we cannot expect that the search for f-score optimal causes is simpler. It seems to be likely that the situation is at least as bad as for recall-optimal causes if not worse.

7. CONCLUSION

In this work we formalized the probability-raising principle in MDPs and studied several quality notions for probability-raising causes. We covered fundamental algorithmic problems for both the strict (local) and global view, where we considered a basic state-based setting in which cause and effect are given as sets of states. We extended this setting to ω-regular path properties as effects in two ways. In a more simple setting we kept causes as sets of states and in a more general approach considered co-safety path properties as causes.

Strict vs. Global probability raising. In our basic setting of state-based cause-effect relations, our results indicate that GPR causes are more general overall by leaving more flexibility to achieve better quality measures, while algorithmic reasoning on SPR causes is simpler. This changed when extending the framework by considering ω-regular effects given by a deterministic Rabin automaton. Our results mainly stem from a polynomial reduction from ω-regular effects to reachability effects (Lemma 6.3). The caveat here is that the strict PR condition translates to a global PR condition after this transformation, which increases the algorithmic complexity of reachability SPR causes to the level of reachability GPR causes. Thus, the strict probability-raising loses its advantage over the global perspective. Furthermore, when considering causes as co-safety path properties we observe

increasing difficulties to handle strict probability-raising. This stems from an underlying problem in the approach of strict probability-raising applied to path properties. As we consider cause-effect relations between these properties, it is somewhat unnatural to require each individual path to raise the probability of the effect property. Rather, it is more natural to say a path property as a whole causes another one, instead of saying all possible realizations of a path property cause another one. This means that co-safety GPR causes also seem more natural than co-safety SPR causes from a philosophical standpoint.

Non-strict inequality in the PR conditions. The approach of probability-raising within this work is in line with the classical notion in literature that uses a strict inequality in the PR condition. As a consequence causes might not exist (see Example 3.5). However, relaxing the PR condition by only requiring a non-strict inequality would apparently be a minor change that broadens the choice of causes. Indeed, the proposed algorithms for checking the SPR and GPR condition for reachability effects (Section 4) can easily be modified for the relaxed definition. As the algorithms of both extended settings discussed in Section 6 stem mainly from a reduction to reachability effects this also holds for reachability and co-safety causes of regular effects. However, a non-strict inequality in the PR condition would lead to a questionable notion of causality, as e.g. {init} would always be a recall- and ratio-optimal cause. Thus, other side constraints are needed in order to make use of the relaxed PR condition. E.g., requiring the non-strict inequality for all schedulers that reach a cause with positive probability and also requiring the existence of a witnessing scheduler for the PR condition with strict inequality might be a useful alternative definition which agrees with Def. 3.1 for Markov chains.

Relaxing the minimality condition. As many causality notions in the literature include some minimality constraint, we included the condition $\Pr_{\mathcal{M}}^{\max}(\neg\mathsf{Cause}\,\mathsf{U}\,\mathsf{c}) > 0$ for all states of Cause in the state-based setting and for reachability PR causes of regular effects. However, this requirement could be dropped without affecting the algorithmic results presented here. This can be useful when the task is to identify components or agents which are responsible for the occurrences of undesired effects. In these cases the cause candidates are fixed (e.g., for each agent $\mathfrak{i}$, the set of states controlled by agent $\mathfrak{i}$), but some of them might violate the minimality condition.

Future directions. In this work we considered type-like causality where cause-effect relations are defined within the model without needing an actual execution that shows the effect. Hence, causes are considered in a forward-looking manner. Notions of probabilistic backward causality that take a concrete execution of the system into account and considerations on PR causality with external interventions as in Pearl's do-calculus [Pea09] are left for future work.

References

[ÁB18]   Erika Ábrahám and Borzoo Bonakdarpour. HyperPCTL: A temporal logic for probabilistic hyperproperties. In Annabelle McIver and András Horváth, editors, *15th International Conference on Quantitative Evaluation of Systems (QEST)*, volume 11024 of *Lecture Notes in Computer Science*, pages 20–35. Springer, 2018. URL: `https://doi.org/10.1007/978-3-319-99154-2_2`.

[BBC+12]   Ilan Beer, Shoham Ben-David, Hana Chockler, Avigail Orni, and Richard J. Trefler. Explaining counterexamples using causality. *Formal Methods in System Design*, 40(1):20–40, 2012.

[BBD+18]   Christel Baier, Nathalie Bertrand, Clemens Dubslaff, Daniel Gburek, and Ocan Sankur. Stochastic shortest paths and weight-bounded properties in Markov decision processes. In Anuj Dawar and Erich Grädel, editors, *33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 86–94. ACM, 2018. URL: `https://doi.org/10.1145/3209108.3209184`.

[BDF+21]   Christel Baier, Clemens Dubslaff, Florian Funke, Simon Jantsch, Rupak Majumdar, Jakob Piribauer, and Robin Ziemek. From verification to causality-based explications (invited talk). In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming, (ICALP)*, volume 198 of *LIPIcs*, pages 1:1–1:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. URL: `https://doi.org/10.4230/LIPIcs.ICALP.2021.1`.

[BFM21]   Christel Baier, Florian Funke, and Rupak Majumdar. A game-theoretic account of responsibility allocation. In Zhi-Hua Zhou, editor, *30th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1773–1779. ijcai.org, 2021. URL: `https://doi.org/10.24963/ijcai.2021/244`.

[BFPZ22]   Christel Baier, Florian Funke, Jakob Piribauer, and Robin Ziemek. On probability-raising causality in Markov decision processes. In Patricia Bouyer and Lutz Schröder, editors, *Foundations of Software Science and Computation Structures*, pages 40–60, Cham, 2022. Springer International Publishing.

[BGC09]   Christel Baier, Marcus Größer, and Frank Ciesinski. Quantitative analysis under fairness constraints. In *International Symposium on Automated Technology for Verification and Analysis*, pages 135–150. Springer, 2009.

[BK08]   Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking (Representation and Mind Series)*. The MIT Press, Cambridge, MA, 2008.

[BKKM14]   Christel Baier, Joachim Klein, Sascha Klüppelholz, and Steffen Märcker. Computing conditional probabilities in Markovian models efficiently. In Erika Ábrahám and Klaus Havelund, editors, *20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, volume 8413 of *Lecture Notes in Computer Science*, pages 515–530. Springer, 2014. URL: `https://doi.org/10.1007/978-3-642-54862-8_43`.

[BT91]   Dimitri P. Bertsekas and John N. Tsitsiklis. An analysis of stochastic shortest path problems. *Mathemathics of Operations Research*, 16(3):580–595, 1991.

[BvH12]   M. Braham and M. van Hees. An anatomy of moral responsibility. *Mind*, 121 (483):601–634, 2012.

[Can88]   John F. Canny. Some algebraic and geometric computations in PSPACE. In *20th Annual ACM Symposium on Theory of Computing (STOC)*, pages 460–467. ACM, 1988.

[CFF+22]   Norine Coenen, Bernd Finkbeiner, Hadar Frenkel, Christopher Hahn, Niklas Metzger, and Julian Siber. Temporal causality in reactive systems. In Ahmed Bouajjani, Lukáš Holík, and Zhilin Wu, editors, *20th International Symposium on Automated Technology for Verification and Analysis, ATVA 2022, October 25-28, 2022, Beijing, China*, 2022.

[CGP99]   E. M. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.

[CH04]   Hana Chockler and Joseph Y. Halpern. Responsibility and blame: A structural-model approach. *J. Artif. Int. Res.*, 22(1):93–115, October 2004.

[CHK08]   Hana Chockler, Joseph Y. Halpern, and Orna Kupferman. What causes a system to satisfy a specification? *ACM Transactions on Computational Logic*, 9(3):20:1–20:26, 2008.

[Cho16]   Hana Chockler. Causality and responsibility for formal verification and beyond. In *First Workshop on Causal Reasoning for Embedded and safety-critical Systems Technologies (CREST)*, volume 224 of *EPTCS*, pages 1–8, 2016. URL: `https://doi.org/10.4204/EPTCS.224.1`.

[CJ20]   Davide Chicco and Giuseppe Jurman. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020.

[dA97]   Luca de Alfaro. *Formal Verification of Probabilistic Systems*. Phd thesis, Stanford University, Stanford, USA, 1997. URL: `https://wcl.cs.rpi.edu/pilots/library/papers/TAGGED/4375-deAlfaro(1997)-FormalVerificationofProbabilisticSystems.pdf`.

[dA99]   Luca de Alfaro. Computing minimum and maximum reachability times in probabilistic systems. In Jos
            C. M. Baeten and Sjouke Mauw, editors, *10th International Conference on Concurrency Theory (CONCUR)*,
            volume 1664 of *Lecture Notes in Computer Science*, pages 66–81. Springer, 1999. URL: `https://doi.org/`
            `10.1007/3-540-48320-9_7`.

[DFT20]   Rayna Dimitrova, Bernd Finkbeiner, and Hazem Torfah. Probabilistic hyperproperties of Markov decision
            processes. In Dang Van Hung and Oleg Sokolsky, editors, *18th International Symposium on Automated
            Technology for Verification and Analysis (ATVA)*, volume 12302 of *Lecture Notes in Computer Science*, pages
            484–500. Springer, 2020. URL: `https://doi.org/10.1007/978-3-030-59152-6_27`.

[Eel91]   Ellery Eells. *Probabilistic Causality*. Cambridge Studies in Probability, Induction and Decision Theory.
            Cambridge University Press, 1991.

[FH19]   Meir Friedenberg and Joseph Y. Halpern. Blameworthiness in multi-agent settings. In *33rd Conference on
            Artificial Intelligence (AAAI)*, pages 525–532. AAAI Press, 2019. URL: `https://doi.org/10.1609/aaai.`
            `v33i01.3301525`.

[FHJ$^+$11] Joan Feigenbaum, James A. Hendler, Aaron D. Jaggard, Daniel J. Weitzner, and Rebecca N. Wright. Account-
            ability and deterrence in online life. In *Proceedings of WebSci '11*, New York, NY, USA, 2011. ACM.

[GJ79]   M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*.
            W. H. Freeman, 1979.

[Hit16]   Christoph Hitchcock. Probabilistic causation. In Alan Hájek and Christopher Hitchcock, editors, *The Oxford
            Handbook of Probability and Philosophy*, pages 815–832. Oxford University Press, 2016.

[HP01]   Joseph Y. Halpern and Judea Pearl. Causes and explanations: A structural-model approach: Part 1: Causes. In
            17th Conference in Uncertainty in Artificial Intelligence (UAI), pages 194–202, 2001.

[Kal20]   Lodewijk Kallenberg. *Lecture Notes Markov Decision Problems - version 2020*. 02 2020.

[Kle12]   Semantha Kleinberg. *Causality, Probability and Time*. Cambridge University Press, 2012.

[KLL11]   Matthias Kuntz, Florian Leitner-Fischer, and Stefan Leue. From probabilistic counterexamples via causality to
            fault trees. In Francesco Flammini, Sandro Bologna, and Valeria Vittorini, editors, *30th International Confer-
            ence on Computer Safety, Reliability, and Security (SAFECOMP)*, volume 6894 of *Lecture Notes in Computer
            Science*, pages 71–84. Springer, 2011. URL: `https://doi.org/10.1007/978-3-642-24270-0_6`.

[KM09]   Samantha Kleinberg and Bud Mishra. The temporal logic of causal structures. In *25th Conference on
            Uncertainty in Artificial Intelligence (UAI)*, pages 303–312, 2009.

[KTK80]   Mikhail K Kozlov, Sergei P Tarasov, and Leonid G Khachiyan. The polynomial solvability of convex quadratic
            programming. *USSR Computational Mathematics and Mathematical Physics*, 20(5):223–228, 1980.

[Lei15]   Florian Leitner-Fischer. *Causality Checking of Safety-Critical Software and Systems*. PhD thesis, University
            of Konstanz, Germany, 2015. URL: `http://kops.uni-konstanz.de/handle/123456789/30778`.

[Lew73]   David Lewis. Counterfactuals and comparative possibility. *Journal of Philosophical Logic*, 2(4):418–446,
            1973.

[Mär20]   Steffen Märcker. *Model checking techniques for design and analysis of future hardware and software systems*.
            PhD thesis, TU Dresden, Germany, 2020. URL: `https://d-nb.info/1232958204`.

[MBKK17]   Steffen Märcker, Christel Baier, Joachim Klein, and Sascha Klüppelholz. Computing conditional probabilities:
            Implementation and evaluation. In Alessandro Cimatti and Marjan Sirjani, editors, *Software Engineering
            and Formal Methods - 15th International Conference, SEFM 2017, Trento, Italy, September 4-8, 2017,
            Proceedings*, volume 10469 of *Lecture Notes in Computer Science*, pages 349–366. Springer, 2017. `doi:`
            `10.1007/978-3-319-66197-1\_22`.

[Mil17]   Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267,
            06 2017.

[MP95]   Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Safety*. Springer-Verlag,
            1995.

[Nam01]   Kedar S. Namjoshi. Certifying model checkers. In *13th International Conference on Computer Aided
            Verification (CAV)*, volume 2102 of *Lecture Notes in Computer Science*, pages 2–13. Springer, 2001. URL:
            `https://doi.org/10.1007/3-540-44585-4_2`.

[PB99]   Stephen D Patek and Dimitri P Bertsekas. Stochastic shortest path games. *SIAM Journal on Control and
            Optimization*, 37(3):804–824, 1999.

[Pea09]   Judea Pearl. *Causality*. Cambridge University Press, 2nd edition, 2009.

[Pow11]   David Powers. Evaluation: From precision, recall and f-measure to ROC, informedness, markedness &
            correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[Put94]   Martin Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley &
              Sons, Inc., New York, NY, 1994.
[PV91]   Panos M Pardalos and Stephen A Vavasis. Quadratic programming with one negative eigenvalue is NP-hard.
              Journal of Global optimization, 1(1):15–22, 1991.
[Rei56]   Hans Reichenbach. *The Direction of Time*. Dover Publications, 1956.
[Sel94]   Alan L. Selman. A taxonomy of complexity classes of functions. *Journal of Computer and System
              Sciences*, 48(2):357–381, 1994. URL: `https://www.sciencedirect.com/science/article/pii/`
              `S0022000005800091`, `doi:https://doi.org/10.1016/S0022-0000(05)80009-1`.
[Sup70]   Patrick Suppes. *A Probabilistic Theory of Causality*. Amsterdam: North-Holland Pub. Co., 1970.
[Vav90]   Stephen A. Vavasis. Quadratic programming is in NP. *Information Processing Letters*, 36(2):73–77, 1990.
              URL: `https://www.sciencedirect.com/science/article/pii/002001909090100C`, `doi:https:`
              `//doi.org/10.1016/0020-0190(90)90100-C`.
[YD16]   Vahid Yazdanpanah and Mehdi Dastani. Distant group responsibility in multi-agent systems. In Matteo Bal-
              doni, Amit K. Chopra, Tran Cao Son, Katsutoshi Hirayama, and Paolo Torroni, editors, *19th International Con-
              ference on Princiles and Practice of Multi-Agent Systems (PRIMA)*, volume 9862 of *Lecture Notes in Computer
              Science*, pages 261–278. Springer, 2016. URL: `https://doi.org/10.1007/978-3-319-44832-9_16`.
[YDJ⁺19]   Vahid Yazdanpanah, Mehdi Dastani, Wojciech Jamroga, Natasha Alechina, and Brian Logan. Strategic
              responsibility under imperfect information. In Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E.
              Taylor, editors, *18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*,
              pages 592–600. International Foundation for Autonomous Agents and Multiagent Systems, 2019. URL:
              `http://dl.acm.org/citation.cfm?id=3331745`.
[ZPF⁺22]   Robin Ziemek, Jakob Piribauer, Florian Funke, Simon Jantsch, and Christel Baier. Probabilistic
              causes in Markov chains. *Innovations in Systems and Software Engineering*, Apr 2022. `doi:10.1007/`
              `s11334-022-00452-8`.