The Satisfiability Problem for Unbounded Fragments of Probabilistic CTL

Jan Křetínský
Technical University of Munich, Germany
jan.kretinsky@tum.de
https://orcid.org/0000-0002-8122-2881

Alexej Rotar
Technical University of Munich, Germany
alexej.rotar@tum.de

Abstract

We investigate the satisfiability and finite satisfiability problem for probabilistic computation-tree logic (PCTL) where operators are not restricted by any step bounds. We establish decidability for several fragments containing quantitative operators and pinpoint the difficulties arising in more complex fragments where the decidability remains open.

2012 ACM Subject Classification Theory of computation → Modal and temporal logics

Keywords and phrases temporal logic, probabilistic verification, probabilistic computation tree logic, satisfiability

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2018.32

Related Version A full version of the paper is available at [24], `https://arxiv.org/abs/1806.11418`.

Funding This research was funded in part by the Czech Science Foundation grant No. P202/12/G061, TUM IGSSE Grant 10.06 (PARSEC), and the German Research Foundation (DFG) project 383882557 "Statistical Unbounded Verification".

1 Introduction

Temporal logics are a convenient and useful formalism to describe behaviour of dynamical systems. Probabilistic CTL (PCTL) [17, 16] is the probabilistic extension of the branching-time logic CTL [12], obtained by replacing the existential and universal path quantifiers with the probabilistic operators, which allow us to quantify the probability of runs satisfying a given path formula. At first, the probabilities used were only 0 and 1 [17], giving rise to the *qualitative PCTL (qPCTL)*. This has been extended to any values from [0, 1] in [16], yielding the *(quantitative) PCTL* (onwards denoted just *PCTL*). More precisely, the syntax of these logics is built upon atomic propositions, Boolean connectives, temporal operators such as $\mathbf{X}$ ("next") and $\mathbf{U}$ ("until"), and the probabilistic quantifier $\bowtie q$ where $\bowtie$ is a numerical comparison such as $\leq$ or $>$, and $q \in [0, 1] \cap \mathbb{Q}$ is a rational constant. A simple example of a PCTL formula is $ok\mathbf{U}_{=1}(\mathbf{X}_{\geq 0.9} finish)$, which says that on almost all runs we reach a state where there is 90% chance to *finish* in the next step and up to this state *ok* holds true. PCTL formulae are interpreted over Markov chains [26] where each state is assigned a subset of atomic propositions that are valid in a given state.

In this paper, we study the *satisfiability problem*, asking whether a given formula has a *model*, i.e. whether there is a Markov chain satisfying it. If a model does exist, we also want to construct it. Apart from being a fundamental problem, it is a possible tool for checking consistency of specifications or for reactive synthesis. The problem has been shown EXPTIME-complete for qPCTL in the setting where we quantify over finite models (*finite satisfiability*) [17, 7] as well as over generally countable models (*infinite satisfiability*) [7]. The problem for (the general quantitative) PCTL remains open for decades. We address this question on fragments of PCTL. The considered fragments are not of primary interst themselves. Rather they illustrate the techniques we develop and how far we can push decidability results when applying only those. In order to get a better understanding of this ultimate problem, we answer the problem for several fragments of PCTL that are

- quantitative, i.e. involving also probabilistic quantification over arbitrary rational numbers (not just 0 and 1),
- step unbounded, i.e. not imposing any horizon for the temporal operators.

Besides, we consider models with unbounded size, i.e. countable models or finite models, but with no a priori restriction on the size of the state space. These are the three distinguishing features, compared to other works. The closest are the following. Firstly, solutions for the qPCTL have been given in [17, 7] and for a more general logic PCTL* in [25, 21]. Secondly, [9] shows decidability for *bounded PCTL* where the scope of the operators is restricted by a step bound to a given time horizon. Thirdly, the *bounded satisfiability problem* is to determine, whether there exists a model of a given size for a given formula. This problem has been solved by encoding it into an SMT problem [4]. There is an important implication of this result. Namely, if we are able to determine a maximum required model size for some formula, then it follows that the satisfiability of that formula can also be determined. We take this approach in some of our proofs. Additionally, we use the result of [7] that the branching degree (number of successors) for a model of a formula ϕ can be bounded by $|\phi| + 2$, where $|\phi|$ is the length of ϕ.

Our contribution is as follows:

- We show decidability of the (finite and infinite) satisfiability problem for several quantitative unbounded fragments of PCTL, focusing on future- and globally-operators ($\mathbf{F}$,$\mathbf{G}$).
- We investigate the relationship between finite and infinite satisfiability on these fragments.
- We identify a fundamental issue preventing us from extending our techniques to the general case. We demonstrate this on a formula enforcing a more complicated form of its models. This allows us to identify the "smallest elegant" fragment where the problem remains open and the solution requires additional techniques.

Due to space constraints, the proofs are sketched and then worked out in detail in [24].

1.1 Further related work

As for the *non-probabilistic* predecessors of PCTL, the satisfiability problem is known to be EXPTIME-complete for CTL [12] as well as the more general modal μ-calculus [3, 15]. Both logics have the small model property [12, 20], more precisely, every satisfiable formula ϕ has a finite-state model whose size is exponential in the size of ϕ. The complexity of the satisfiability problems has been investigated also for fragments of CTL [23] and the modal μ-calculus [18].

The satisfiability problem for qPCTL and qPCTL* was investigated already in the early 80's [25, 21, 17], together with the existence of sound and complete axiomatic systems. The decidability for qPCTL over countable models also follows from these general results for qPCTL*, but the complexity was not examined until [7], showing it is also EXPTIME-complete, both for finite and infinite satisfiability.

While the decidability of satisfiability is open, there are only few negative results. [7] proves *undecidability* of the problem whether for a given PCTL formula there exists a model with a branching degree that is bounded by a given integer, where the branching degree is the number of successors of a state. However, the authors have not been able to extend their proof and show the undecidability for the general problem.

The PCTL *model checking problem* is the task to determine, whether a given system satisfies a given formula, i.e. whether it is a model of the formula. This problem has been studied both for finite and infinite Markov chains and decision processes, see e.g. [10, 19, 14, 13, 8]. The PCTL *strategy synthesis* problem asks whether the non-determinism in a given Markov decision process can be resolved so that the resulting Markov chain satisfies the formula [1, 22, 5, 6].

2 Preliminaries

In this section, we recall basic notions related to (discrete-time) Markov chains [26] and the probabilistic CTL [16]. Let $\mathcal{A}$ be a finite set of atomic propositions.

2.1 Markov chains

▶ **Definition 1** (Markov chain). A *Markov chain* is a tuple $M = (S, P, s_0, L)$ where S is a countable set of *states*, $P : S \times S \to [0, 1]$ is the *probability transition matrix* such that, for all $s \in S$, $\sum_{t \in S} P(s, t) = 1$, $s_0 \in S$ is the *initial* state, and $L : S \to 2^{\mathcal{A}}$ is a labeling function.

Whenever we write M, we implicitly mean a Markov chain (S, P, s_0, L). The semantics of a Markov chain M, is the probability space $(\mathit{Runs}_M, \mathcal{F}_M, \mathbb{P}_M)$, where $\mathit{Runs}_M = S^\omega$ is the set of *runs* of M, $\mathcal{F}_M \subseteq 2^{S^\omega}$ is the σ-algebra generated by the set of cylinders of the form $\mathit{Cyl}_M(\rho) = \{\pi \in S^\omega \mid \rho \text{ is a prefix of } \pi\}$ and the probability measure is uniquely determined [2] by $\mathbb{P}_M(\mathit{Cyl}_M(\rho_0 \cdots \rho_n)) := \prod_{0 \leq i < n} P(\rho_i, \rho_{i+1})$ if $\rho_0 = s_0$ and 0 otherwise.

We say that a state is *reached* on a run if it appears in the sequence; a set of states is reached if some of its states are reached. The immediate successors of a state s are denoted by $\mathit{post}_M(s) := \{t \in S \mid P(s, t) > 0\}$ and the set of states reachable with positive probability is the reflexive and transitive closure $\mathit{post}^*_M(s)$. We will write $\mathbb{P}(\cdot)$, $\mathit{post}(\cdot)$, and $\mathit{post}^*(\cdot)$, if M is clear from the context.

The *unfolding* of a Markov chain M is the Markov chain $T_M := (S^+, P', s_0, L')$ with the form of an infinite tree given by $P'(\rho s, \rho s s') = P(s, s')$ and $L'(\rho s) = L(s)$. Each state of T_M maps naturally to a state of M (the last one in the sequence), inducing an equivalence relation $\rho s \sim \rho s'$ iff $s = s'$. Consequently, each run of T_M maps naturally to a run of M and the unfolding preserves the measure of the respective events.

For a Markov chain M, a set $T \subseteq S$ is called *strongly connected* if for all $s, t \in T$, $t \in \mathit{post}^*(s)$; it is a *strongly connected component (SCC)* if it is maximal (w.r.t. inclusion) with this property. If, moreover, $\mathit{post}^*(t) \subseteq T$ for all $t \in T$ then it is a *bottom SCC (BSCC)*. A classical result, see e.g. [2], states that the set of states visited infinitely often is almost surely, i.e. with probability 1, a BSCC:

▶ **Lemma 2.** *In every finite Markov chain, the set of BSCCs is reached almost surely. Further, conditioning on runs reaching a BSCC C, every state of C is reached infinitely often almost surely.*

2.2 Probabilistic Computational Tree Logic

The definition of probabilistic CTL (PCTL) [16] is usually based on the next- and until-operators ($\mathbf{X}$, $\mathbf{U}$). In this paper, we restrict our attention to the future- and globally operators ($\mathbf{F}$, $\mathbf{G}$), which can be derived from the until-operator. Further, w.l.o.g. we impose the negation normal form and the lower-bound-comparison normal form; for the respective transformations see, e.g., [2].

▶ **Definition 3** (PCTL($\mathbf{F}$,$\mathbf{G}$) syntax and semantics)**.** The *formulae* are given by the following syntax:

$$\Phi ::= a \mid \neg a \mid \Phi \wedge \Phi \mid \Phi \vee \Phi \mid \mathbf{F}_{\rhd q}\Phi \mid \mathbf{G}_{\rhd q}\Phi$$

where $q \in [0,1]$, $\rhd \in \{\geq, >\}$, and $a \in \mathcal{A}$ is an atomic proposition. Let M be a Markov chain and $s \in S$ its state. We define the modeling relation $\models$ inductively as follows

M1 $M, s \models a$ iff $a \in L(s)$
M2 $M, s \models \neg a$ iff $a \notin L(s)$
M3 $M, s \models \phi \wedge \psi$ iff $M, s \models \phi$ and $M, s \models \psi$
M4 $M, s \models \phi \vee \psi$ iff $M, s \models \phi$ or $M, s \models \psi$
M5 $M, s \models \mathbf{F}_{\rhd q}\varphi$ iff $\mathbb{P}_{M(s)}(\{\pi \mid \exists i \in \mathbb{N}_0 : M, \pi[i] \models \varphi\}) \rhd q$
M6 $M, s \models \mathbf{G}_{\rhd q}\varphi$ iff $\mathbb{P}_{M(s)}(\{\pi \mid \forall i \in \mathbb{N}_0 : M, \pi[i] \models \varphi\}) \rhd q$
where $M(s)$ is M with s being the initial state, and $\pi[i]$ is the ith element of π. We say that M is a *model* of φ if $M, s_0 \models \varphi$.

We will denote the set of literals by $\mathcal{L} := \mathcal{A} \cup \{\neg a \mid a \in \mathcal{A}\}$. Instead of the constraint ≥ 1, we often write $= 1$. Further, we define the set of all subformulae. This definition slightly deviates from the usual definition of subformulae, e.g. the one in [7], in that $\neg a \in sub(\phi)$ does not necessarily imply $a \in sub(\phi)$.

▶ **Definition 4** (Subformulae)**.** The set $sub(\phi)$ is recursively defined as follows

- $\phi \in sub(\phi)$
- if $\psi \wedge \xi \in sub(\phi)$ or $\psi \vee \xi \in sub(\phi)$, then $\psi, \xi \in sub(\phi)$
- if $\mathbf{F}_{\rhd q}\psi \in sub(\phi)$ or $\mathbf{G}_{\rhd q}\psi \in sub(\phi)$ then $\psi \in sub(\phi)$

Next, we introduce the satisfiability problems, which are the main topic of the paper.

▶ **Definition 5** (The satisfiability problems)**.** A formula ϕ is called *(finitely) satisfiable*, if there is a (finite) model for ϕ. Otherwise, it is (finitely) unsatisfiable. The (finite) satisfiability problem is to determine whether a given formula is (finitely) satisfiable.

Instead of simply writing "satisfiable" we sometimes stress the absence of "finitely" and write "generally satisfiable" for satisfiablity on countable, i.e. finite or countably infinite, models. For some proofs, it is more convenient to consider the unfolding of a Markov chain instead of the original one. As we mentioned already, the measure of events is preserved in the unfolding of a chain. Hence, we can state the following lemma.

▶ **Lemma 6.** *If M is a model of ϕ then its unfolding T_M is a model of ϕ.*

We say that formulae ϕ, ψ are (finitely) equivalent if they have the same set of (finite) models, written $\phi \equiv \psi$ ($\phi \equiv_{fin} \psi$); that they are (finitely) equisatisfiable if they are both (finitely) satisfiable or both (finitely) unsatisfiable; and that $\phi \Rightarrow \psi$ if every model of ϕ is also a model of ψ.

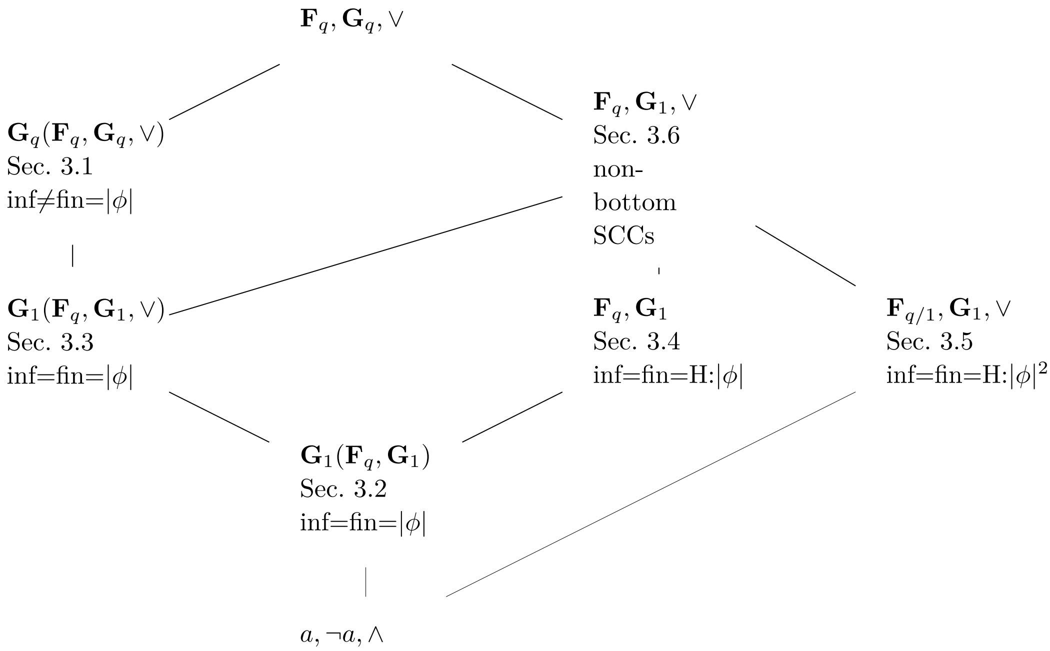

Figure 1 Hasse diagram summarizing the satisfiability results for the considered fragments of PCTL($\mathbf{F}$, $\mathbf{G}$), all containing literals and conjunctions, and some form of quantitative comparisons. The fragments are described by the list of operators they allow (excluding the constructs of the minimal fragment). The subscript denotes the possible constraints on probabilistic operators. $\mathbf{G}_x(\mathit{list})$ denotes formulae in the fragment described by *list* with $\mathbf{G}$-operators at the top-level. fin and inf abbreviate finite and general satisfiability, respectively. fin=inf denotes that the problems are equivalent. H:x denotes that the height of a tree model can be bounded by x. By $= x$ we denote that the model size can be bounded by x. The $\mathbf{F}_q, \mathbf{G}_1, \vee$-fragment might require non-bottom SCCs in finite models.

3 Results

In this section we present our results. A summary is schematically depicted in Fig. 1. We briefly describe the considered fragments; the full formal definitions can be found in the respective sections. Since already the satisfiability for propositional logic in negation normal form has nontrivial instances only when all the constructs $a, \neg a$ and conjunction are present, we only consider fragments with all three included; see the bottom of the Hasse diagram. The fragments are named by the list of constructs they use, where we omit the three constructs above to avoid clutter. Here 1 stands for ≥ 1 and q stands for $\rhd q$ for all $q \in [0, 1] \cap \mathbb{Q}$. Further, $\mathbf{G}_x(\mathit{list})$ denotes the sub-fragment of *list* where the topmost operator is $\mathbf{G}_x$. Finally, $\mathbf{F}_{q/1}$ denotes the use of $\mathbf{F}_q$ with the restriction that inside $\mathbf{G}$ only $q = 1$ can be used.

The fragments are investigated in the respective sections. We examine the problems of the general satisfiability ("inf") and the finite satisfiability ("fin"); equality denotes the problems are equivalent. We use two results to prove decidability of the problems. Firstly, [4] shows that given a formula ϕ and an integer n, one can determine whether or not there is a model for ϕ that has at most n states. Consequently, we obtain the decidability result whenever we establish an upper bound on the size of smallest models. Here "$|\phi|$" denotes the satisfiability of a given ϕ on models of size $\leq |\phi|$. Secondly, [7] establishes that for any satisfiable PCTL formula there is a model with branching bounded by $|\phi| + 2$. Consequently, we obtain the decidability result whenever it is sufficient to consider trees of a certain height H (with back edges) since the number of their nodes is then bounded by $(|\phi| + 2)^H$. Here "H:n" denotes that the models can be limited to a height $H \leq n$.

While we obtain decidability in the lower part of the diagram, the upper part only treats finite satisfiability, and in particular for $\mathbf{F}_q, \mathbf{G}_1, \vee$, we only demonstrate that models with more complicated structure are necessary. Namely, the models may be of unbounded sizes for structurally same formulae – i.e. formulae which only differ in the constraints on the temporal operators – or require presence of non-bottom SCCs, see Section 3.6 and the discussion in Section 4.

3.1 Finite satisfiability for $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$

This section treats $\mathbf{G}$-formulae of the $\mathbf{F}_q, \mathbf{G}_q$-fragment, i.e. of PCTL($\mathbf{F}, \mathbf{G}$). In particular, it includes $\mathbf{G}_{>0}$-formulae. In general, formulae in this fragment (even without quantified $\mathbf{F}$ and $\mathbf{G}$-operators) can enforce rather complicated behaviour [7]. Therefore, we will focus on finitely satisfiable formulae. We will see that they can be satisfied by rather simple models.

▶ **Definition 7.** $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$-formulae are given by the grammar

$$\Phi ::= \mathbf{G}_{\rhd q}\Psi$$
$$\Psi ::= a \mid \neg a \mid \Psi \wedge \Psi \mid \Psi \vee \Psi \mid \mathbf{F}_{\rhd q}\Psi \mid \mathbf{G}_{\rhd q}\Psi$$

The main result of this section is that finitely satisfiable formulae in this fragment can be satisfied by models of size linear in $|\phi|$.

▶ **Theorem 8.** *Let ϕ be a finitely satisfiable $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$-formula. Then ϕ has a model of size at most $|\phi|$.*

Intuitively, we obtain the result from the fact that some BSCC is reached almost surely and every state in a BSCC is reached almost surely, once we have entered one. In infinite models, BSCCs are not reached almost surely and therefore the proofs cannot be extended to general satisfiability. The following lemma and its proof demonstrate how we can make use of the BSCC properties in order to obtain an equisatisfiable formula in a simpler fragment.

▶ **Lemma 9.** *Let ϕ be a $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$-formula. Then, ϕ is finitely equisatisfiable to a $\mathbf{G}_1(\mathbf{F}_1, \mathbf{G}_1)$-formula ϕ', such that $\phi' \Rightarrow \phi$.*

Proof Sketch. Write ϕ as $\mathbf{G}_{\rhd q}\psi$. Assume that we have a finite model M for $\mathbf{G}_{\rhd q}\psi$. Intuitively, we can select a BSCC that satisfies $\mathbf{G}_{=1}\psi$. We know that there is a BSCC because we are dealing with a finite model. We also know that there is at least one BSCC satisfying our formula, for otherwise M would not be a model for it. In a BSCC, every state is reached almost surely from every other state by Lemma 2. Hence, we can select exactly one state for each $\mathbf{F}$-subformula which satisfies that formula's argument. Then we can create a new BSCC from these states, arranging them, e.g., in a circle. This BSCC models $\mathbf{G}_{=1}\hat{\psi}$, where $\hat{\psi}$ replaces all probabilistic operators with their "almost surely" version. Hence, we have created a model for a $\mathbf{G}_1(\mathbf{F}_1, \mathbf{G}_1)$ formula from a model for $\mathbf{G}_{\rhd q}\psi$. The opposite direction follows from the fact that $\mathbf{G}_{=1}\hat{\psi} \Rightarrow \mathbf{G}_{\rhd q}\psi$. ◀

Note that the transformation does not produce an equivalent formula. Hence, we cannot replace an occurrence of such a formula in a more complex formula. For instance, the formula $\mathbf{G}_{\geq 1/2}\neg a \wedge \mathbf{F}_{\geq 1/2}a$ is satisfiable, whereas $\mathbf{G}_{=1}\neg a \wedge \mathbf{F}_{\geq 1/2}a$ is not. The proof does not work for equality because we are selecting one BSCC while ignoring the rest. This example

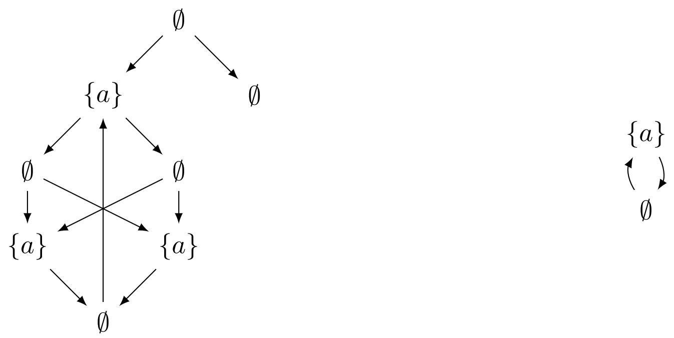

Figure 2 A large and a small model for Formula (1).

demonstrates why we cannot ignore certain BSCCs in general. Using the above result, it is easy to prove Theorem 8.

Proof Sketch. [Proof Sketch of Theorem 8] This follows immediately from the proof of Lemma 9. The BSCC that we have created has at most as many states as there are $\mathbf{F}$-subformulae, which is bounded by $|\phi|$. ◀

▶ **Example 10.** Consider the formula

$$\phi := \mathbf{G}_{\geq 1/2}(\mathbf{F}_{\geq 1/3} a \wedge \mathbf{F}_{\geq 1/3} \neg a). \tag{1}$$

The large Markov chain in Figure 2 models ϕ. Unlabeled arcs indicate a uniform distribution over all successors. It is clear that the model is unnecessarily complicated. After reducing it according to Lemma 9, we obtain the smaller Markov chain on the right.

The example below shows that satisfiability is not equivalent to finite satisfiability for this fragment, and that the proposed transformation does not preserve equisatisfiability over general models. The decidability of the general satisfiability thus remains open here.

▶ **Example 11.** Note that we made use of the BSCC properties for the proofs of this subsection, such as that some BSCC is reached almost surely. Since this is only the case for finite Markov chains, our transformation only holds for finite satisfiability. If we consider the general satisfiability problem, then the equivalent of Lemma 9 is not true. For instance, the formula

$$\phi := \mathbf{G}_{>0}(\neg a \wedge \mathbf{F}_{>0} a) \tag{2}$$

is satisfiable, but requires infinite models, as pointed out in [7]. One such model is given in Figure 3. Observe that the single horizontal run has measure greater than 0. Now consider

$$\hat{\phi} := \mathbf{G}_{=1}(\neg a \wedge \mathbf{F}_{=1} a)$$

Obviously, this formula unsatisfiable. Hence, in this case ϕ is not equisatisfiable to $\hat{\phi}$.

3.2 Satisfiability for $\mathrm{G}_1(\mathrm{F}_q, \mathrm{G}_1)$

This section treats $\mathbf{G}$-formulae of a fragment where $\mathbf{G}_{\rhd q}$ only appears with $q = 1$ and there is no disjunction. The results are later utilized in a richer fragment in Section 3.4. In fact, the main result of this section is an immediate consequence of the main theorem of Section 3.3. Still, the results are interesting themselves as they show some properties of models for formulae in this fragment which do not apply in the generalized case.

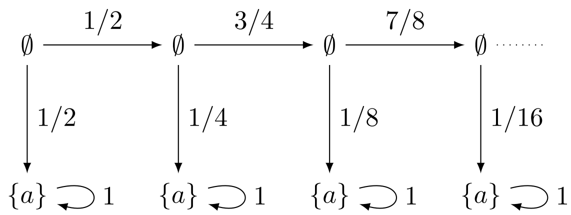

Figure 3 An infinite model for Formula (2).

▶ **Definition 12.** $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formulae are given by the grammar

$$\Phi ::= \mathbf{G}_{=1}\Psi$$
$$\Psi ::= a \mid \neg a \mid \Psi \wedge \Psi \mid \mathbf{F}_{\triangleright q}\Psi \mid \mathbf{G}_{=1}\Psi$$

We prove that satisfiable formulae of this fragment are satisfiable by models of linear size and thus also finitely satisfiable.

▶ **Theorem 13.** *Let ϕ be a satisfiable $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formula. Then ϕ has a model of size at most $|\phi|$.*

The idea here is that we can find a state which behaves similarly to a BSCC (even in infinite models) in that it satisfies all $\mathbf{G}$-subformulae. We can then use this state's successors to construct a small model. The outline of the proof is roughly as follows: First we show that from every state and for every subformula we can find a successor that satisfies this subformula. Using this, we can show that there is a state that satisfies all $\mathbf{G}$-subformulae.

▶ **Lemma 14.** *Let ϕ be a satisfiable $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formula and M its model. Then, for every $\psi \in sub(\phi)$, and $s \in S$, there is a state $t \in post^*(s)$, such that $M, t \models \psi$.*

Proof Sketch. This follows from the fact that we do not allow disjunctions in this fragment. We apply induction over the depth of a subformula ψ. If the formula is ϕ itself, then there is nothing to show. Otherwise, the induction hypothesis yields that the higher-level subformulae are satisfied at some state s. From this, we can easily see that in all possible cases the claim follows: If the higher-level formula is a conjunction, then ψ is one of its conjuncts. Since both conjuncts must be satisfied by s, in particular ψ must be satisfied at s. A similar argument applies to $\mathbf{G}$-formulae. If it is of the form $\mathbf{F}_{\triangleright q}\xi$, then we know that there must be a reachable state where ξ holds. ◀

This concludes the first part of the proof. We continue with the second part and prove that we can find a state which satisfies all $\mathbf{G}$-subformulae.

▶ **Lemma 15.** *Let ϕ be a satisfiable $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formula, M its model, and let $G := \{\psi \in sub(\phi) \mid \psi = \mathbf{G}_{=1}\xi$ for some $\xi\}$. Then there is a state $s \in S$ such that $M, s \models \psi$ for all $\psi \in G$.*

Proof Sketch. It is clear that after encountering a $\mathbf{G}$-formula at some state, all successors will also satisfy it. Therefore, the set of satisfied $\mathbf{G}$-formulae is monotonically growing and bounded. Hence, we can apply induction over the number of yet unsatisfied $\mathbf{G}$-formulae. In every step, we are looking for the next state to satisfy an additional $\mathbf{G}$-formula. This is always possible (as long as there are still unsatisfied ones), due to Lemma 14. ◀

Now, we can prove Theorem 13.

Proof Sketch of Theorem 13. By Lemma 15, we can find a state that satisfies all $\mathbf{G}$-subformulae. In some sense, this state's subtree resembles a BSCC. We can include exactly one state for each $\mathbf{F}$-subformula and create a BSCC out of those states, e.g., arrange them in a circle. We apply induction over $\psi \in \mathit{sub}(\phi)$. The satisfaction of literals and conjunctions is straightforward. Since every state is reached almost surely, every $\mathbf{F}$-formula will be satisfied that way. The satisfaction of the $\mathbf{G}$-formulae follows from the fact that all states used to satisfy all $\mathbf{G}$-formulae in the original model, and from the induction hypothesis. ◀

For the case of finite satisfiability, we also present an alternative proof, which sheds more light on this fragment and its super-fragments. For details, see [24, Appendix B]. Let $\equiv_{\mathit{fin}}$ denote equivalence of PCTL formulae over finite models.

▶ **Theorem 16.** *Let ϕ be a $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formula. Then, the following equivalence holds:*

$$\mathbf{G}_{=1}\phi \equiv_{\mathit{fin}} \mathbf{G}_{=1}\big(\bigwedge_{l \in A} l \wedge \mathbf{F}_{=1}\mathbf{G}_{=1} \bigwedge_{l \in B} l \wedge \bigwedge_{i \in I} \mathbf{F}_{=1} \bigwedge_{l \in C_i} l\big)$$

for appropriate $I \subset \mathbb{N}$, and $A, B, C_i \subset \mathcal{L}$.

Proof Sketch. The proof is based on the following auxiliary statements

$$\mathbf{G}_{=1}\mathbf{G}_{=1}\phi \equiv \mathbf{G}_{=1}\phi \tag{3}$$

$$\mathbf{G}_{=1}\mathbf{F}_{\triangleright q}\phi \equiv_{\mathit{fin}} \mathbf{G}_{=1}\mathbf{F}_{=1}\phi \tag{4}$$

$$\mathbf{F}_{=1}\mathbf{F}_{\triangleright q}\phi \equiv \mathbf{F}_{\triangleright q}\phi \tag{5}$$

and follows by induction.

The second statement is the most interesting one. Intuitively, it is a zero-one law, stating that infinitely repeating satisfaction with a positive probability ensures almost sure satisfaction. Notably, this only holds if the probabilities are bounded from below, hence for finite models, not necessarily for infinite models. ◀

It is an easy corollary of this theorem that a satisfiable formula has a model of a circle form with A and B holding in each state and each element in each C_i holding in some state. In general the models can be of a lasso shape where the initial (transient) part only has to satisfy A, allowing for easy manipulation in extensions of this fragment.

▶ Remark. Note that the equivalence does not hold over infinite models. Indeed, consider as simple a formula as $\mathbf{G}_{=1}\mathbf{F}_{>0}a$, which is satisfied on the Markov chain of Fig. 3 [7], while this does not satisfy the transformed $\mathbf{G}_{=1}\mathbf{F}_{=1}a$. Crucially, equivalence (4) does not hold already for this tiny fragment. Interestingly, when we build a model for the transformed formula, which is equisatisfiable but not equivalent, it turns out to be a model of the original formula. If, moreover, we consider $\neg a, \wedge, \mathbf{G}_{>0}$ then finite and general satisfiability start to differ.

Before we move on to the next fragment, we will prove another consequence of Lemma 14. It is a statement about the BSCCs of models for formulae in this fragment and will be used later for the proof of Theorem 21.

▶ **Corollary 17.** *Let $\mathbf{G}_{=1}\phi$ be a satisfiable $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1)$-formula and M its model. Then, for every BSCC $T \subseteq \mathit{post}^*(s_0)$ of M, the following holds*

1. *For all $\psi \in \mathit{sub}(\mathbf{G}_{=1}\phi)$, there is a state $t \in T$, such that $M, t \models \psi$.*
2. *For all $\mathbf{G}_{=1}\psi \in \mathit{sub}(\mathbf{G}_{=1}\phi)$, and for all states $t \in T$, $M, t \models \mathbf{G}_{=1}\psi$.*

Proof. Point 1 follows from the fact that every reachable BSCC must satisfy $\mathbf{G}_{=1}\phi$, and from Lemma 14. Point 2 follows immediately from point 1. ◀

Note that we did not assume finite satisfiability here, so the model might not contain a single BSCC. In that case, the claim is trivially true. However, Theorem 13 allows us to focus on finitely satisfiable formulae in this fragment.

3.3 Satisfiability for $\mathbf{G_1(F_q, G_1, \vee)}$

This section treats $\mathbf{G}$-formulae of the fragment where $\mathbf{G}_{\rhd q}$ only appears with $q = 1$. We thus lift a restriction of the previous fragment and allow for disjunctions. We generalize the obtained results to this larger fragment. We mentioned earlier that some of the results of the previous fragment do not apply here. Concretely, Lemma 14 does not hold here; that is, there might be subformulae which are not satisfied almost surely. Therefore, there is not necessarily a state that satisfies all $\mathbf{G}$-subformulae. For example, consider $\mathbf{G}_{=1}(\mathbf{F}_{>0}\mathbf{G}_{=1}a \vee \mathbf{F}_{>0}\mathbf{G}_{=1}\neg a)$. There cannot be a single BSCC to satisfy both disjuncts. Although this is not a problem for the results of this section, it will turn out to be a fundamental problem when dealing with arbitrary formulae of the $\mathbf{F}_q, \mathbf{G}_1, \vee$ fragment.

▶ **Definition 18.** $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1, \vee)$-formulae are given by the grammar

$$\Phi ::= \mathbf{G}_{=1}\Psi$$
$$\Psi ::= a \mid \neg a \mid \Psi \wedge \Psi \mid \Psi \vee \Psi \mid \mathbf{F}_{\rhd q}\Psi \mid \mathbf{G}_{=1}\Psi$$

We prove that satisfiable formulae of this fragment are satisfiable by models of linear size and thus also finitely satisfiable.

▶ **Theorem 19.** *Let ϕ be a satisfiable $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1, \vee)$-formula. Then ϕ has a model of size at most $|\phi|$.*

Proof Sketch. The proof for this theorem works essentially the same as it did for Theorem 13. Recall that we looked for a state to satisfy all $\mathbf{G}$-formulae. Though we will not necessarily find a state that does so in this fragment, we can look for a state that satisfies maximal subsets of satisfied $\mathbf{G}$-formulae. Then, we can continue in a similar way as we did in the simpler setting. ◀

3.4 Satisfiability for $\mathbf{F_q, G_1}$

This section treats general formulae of the fragment with no disjunction and where $\mathbf{G}_{\rhd q}$ only appears with $q = 1$.

▶ **Definition 20.** $\mathbf{F}_q, \mathbf{G}_1$-formulae are given by the grammar

$$\Phi ::= a \mid \neg a \mid \Phi \wedge \Phi \mid \mathbf{F}_{\rhd q}\Phi \mid \mathbf{G}_{=1}\Phi$$

In Section 3.2 we discussed a special case of this fragment, where the top-level operator is $\mathbf{G}_{=1}$. Two results are particularly interesting for this section: Firstly, the construction of models for such formulae as explained in the proof of Theorem 13, and secondly, the properties of BSCCs in models for such formulae as stated in Corollary 17. We will use those in order to simplify models in this generalized setting. We say a Markov chain has *height h* if it is a tree with back edges of height h.

▶ **Theorem 21.** *A satisfiable $\mathbf{F}_q, \mathbf{G}_1$-formula ϕ has a model of height $|\phi|$.*

Proof Sketch. Our aim is to transform a given, possibly infinite model into a tree-like shape. To do so, we first construct a tree by considering all $\mathbf{F}$-formulae which are not nested in $\mathbf{G}$s (in the following simply non-nested $\mathbf{F}$-formulae). Each path in this tree will satisfy each of these $\mathbf{F}$-formulae at most once. At the end of each path, we will then insert BSCCs satisfying the $\mathbf{G}$-formulae, in the spirit of Theorem 13.

The collapsing procedure from a state s is as follows: We first determine which of the $\mathbf{F}$-formulae that are satisfied at s are relevant. Those are the formulae which are not nested in other temporal formulae and have not yet been satisfied on the current path. Once we have determined this set, say I, we need to find the successors which are required to satisfy the formulae in I. For this, we construct the set $sel(s)$. Informally, $sel(s)$ contains all states t s.t. (i) t satisfies at least one formula $\psi \in I$, and (ii) there is no state on the path between s and t satisfying ψ. Formally, $sel(s) := \{t \in post^*(s) \mid \exists \mathbf{F}_{\triangleright r}\psi \in I.\ M, t \models \psi \wedge \forall t' \in post^*(s) \cap pre^*(t). M, t \not\models \psi\}$. Then, we connect s to every state in $sel(s)$ directly; i.e. for $t \in sel(s)$, we set $P(s,t) := P^*(s,t)$, and for all states $t \notin sel(s)$, we set $P(s,t) := 0$.[1] A simple induction on the length of a path yields that every state that is reachable from s in the constructed MC is reached with at least the probability as in the original one. From this, one can easily see that every non-nested $\mathbf{F}$-formula is satisfied. The new set $post(s)$ might be infinite. However, we know that we can prune most of the successors and limit the branching degree to $|\phi| + 2$ [7]. Then, we repeat the procedure from each of the successors. Since the number of non-nested $\mathbf{F}$-formulae decreases with every step, we will reach states which do not have to satisfy non-nested $\mathbf{F}$-formulae at all on every branch. The number of steps we need to reach such states, is bounded by the number of non-nested $\mathbf{F}$-formulae in ϕ. At those states, we can use Theorem 13 to obtain models for the respective $\mathbf{G}$-formulae. Those are of size linear in the size of the $\mathbf{G}$-formulae. The overall height is then bounded by $|\phi|$. The fact that the resulting MC is a model can be easily proved by induction over $|\phi|$. ◀

The models that we construct have a quite regular shape: They start as a tree and in every step ensure satisfaction of one of the $\mathbf{F}$-formulae. As soon as they have satisfied all outer $\mathbf{F}$-formulae, on every branch a model of circle shape for the respective $\mathbf{G}$-formula follows. Since the branching degree is at most $|\phi| + 2$ and the number of steps before we repeat a state is bounded by $|\phi|$, the overall size is bounded to $(|\phi| + 2)^{2|\phi|+1}$.

▶ **Example 22.** Let $\phi^p := \mathbf{F}_{\geq p}\mathbf{G}_{=1}a$, and $\psi^p := \mathbf{F}_{\geq p}\mathbf{G}_{=1}\neg a$. The large Markov chain of Figure 4 is a model for $\phi^{1/2} \wedge \psi^{1/2}$. The grayed states illustrate the set $sel(s_0)$. The other boxes show the sets $sel(.)$ of the respective grayed state. Everything in between is omitted. The smaller Markov chain is the reduced version of the original model.

3.5 Satisfiability for $\mathbf{F}_{q/1}, \mathbf{G}_1, \vee$

In the previous section we have been able to construct simple models for formulae of the $\mathbf{F}_q, \mathbf{G}_1$-fragment by exploiting the nature of $\mathbf{G}$-formulae thereof as presented in Section 3.2. This works because every formula nested within a $\mathbf{G}$ is satisfied in every BSCC. Hence, we can simply postpone the satisfaction of those until we reach a BSCCs. In the $\mathbf{F}_q, \mathbf{G}_1, \vee$-fragment, this is not the case anymore, as discussed in Section 3.3. This can cause some complications, which are discussed in more detail in Section 3.6. In order to be able to apply similar techniques as in the previous section, we can simplify the fragment and enforce the property that $\mathbf{F}$-formulae occur only with $q = 1$ within $\mathbf{G}$s.

[1] In fact, we need to scale $P(s,t)$ in order to obtain a Markov chain, in general, as the probability to reach $sel(s)$ might be less than 1. For details, refer to the formal proof in [24, Appendix A].

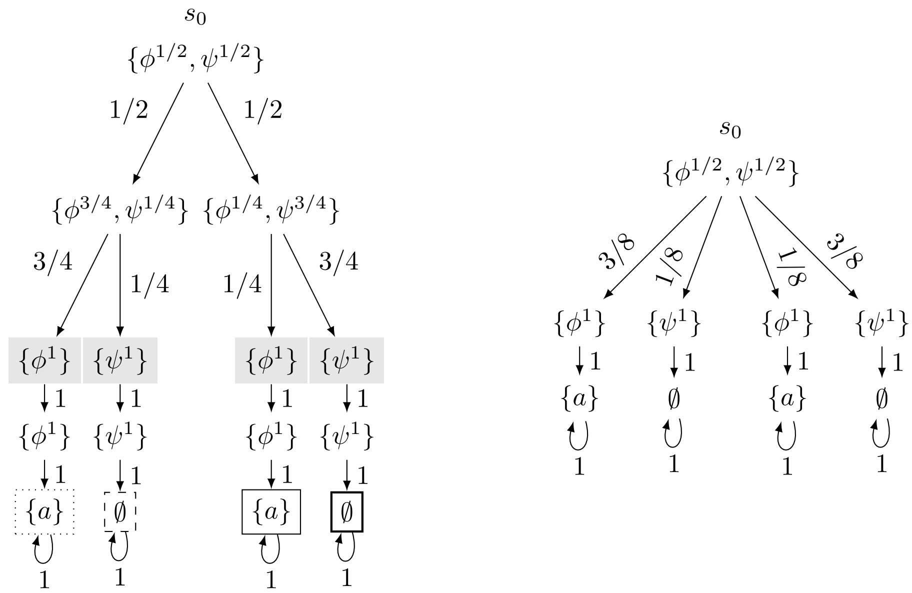

Figure 4 Example of a reduction for a $\mathbf{F}_q, \mathbf{G}_1$-formula.

▶ **Definition 23.** $\mathbf{F}_{q/1}, \mathbf{G}_1, \vee$-formulae are given by the grammar

$$\Phi ::= a \mid \neg a \mid \Phi \wedge \Phi \mid \Phi \vee \Phi \mid \mathbf{F}_{\rhd q}\Phi \mid \mathbf{G}_{=1}\Psi$$
$$\Psi ::= a \mid \neg a \mid \Psi \wedge \Psi \mid \Psi \vee \Psi \mid \mathbf{F}_{=1}\Psi \mid \mathbf{G}_{=1}\Psi$$

Again, we show that the necessary minimal height of models can be bounded.

▶ **Theorem 24.** *A satisfiable $\mathbf{F}_{q/1}, \mathbf{G}_1, \vee$-formula ϕ has a model of height $|\phi|^2$.*

Proof Sketch. As the first step, we apply the same procedure as in the proof of Theorem 21. The outer $\mathbf{F}$-formulae are then satisfied for the same reason as in the setting without disjunctions. However, the $\mathbf{G}$-nested $\mathbf{F}$-formulae might not be satisfied anymore because the BSCCs do not necessarily satisfy each of them. Since the $\mathbf{G}$-nested $\mathbf{F}$-formulae appear only with $q = 1$, we know that once a state of the original model satisfies such a formula, almost every path satisfies the respective path formula. Let s be a state of the reduced chain, and $t \in post(s)$. In the original model, there might be states in between s and t. If some $\mathbf{G}$-nested $\mathbf{F}$-formula (say $\mathbf{F}_{=1}\psi$) which is satisfied at s is also satisfied at t, we do not need to take care of it. If this is not the case, then we know for sure that some of the states between s and t must satisfy ψ. We can determine such states for each $\mathbf{G}$-nested $\mathbf{F}$-formula. We include exactly one of those for each such formula. Then, preserving the order, we chain them in such a way that each one has a unique successor. The last one's unique successor is t. Let s' be the first one. Then, we set $P(s, s') := P(s, t)$. We repeat this procedure for each state of the reduced chain.

However, we must be careful when picking the states that we want to include. For instance, if we always pick the first state to satisfy some formula, then we might in the end violate ϕ. Say we have two formulae $\mathbf{F}_{=1}\psi$ and $\mathbf{F}_{=1}\xi$ and states u, v, w occuring in this order, such that $M, u \models \psi$, $M, v \models \xi$, and $M, w \models \psi$. If $M, v \models \mathbf{F}_{=1}\psi$ in the original model, we might violate it if we omit w. Therefore, we shall rather omit u und pick w. In general, we pick the last possible state for each formula. In a sense, we look backwards starting from t. This way, it is guaranteed that whenever we have not yet satisfied some $\mathbf{F}$-formula, it will be propagated to t.

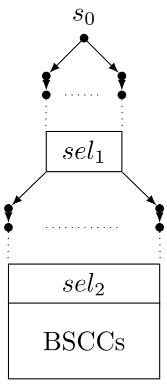

Figure 5 Reduction of models for $\mathbf{F}_{q/1}, \mathbf{G}_1, \vee$.

This way, we preserve the reachability probabilities and therefore the satisfaction of the outer **F**-formulae. The newly added states guarantee the satisfaction of nested **F**-formulae. An induction over ϕ shows that the constructed MC is again a model. Since we add at most $|\phi|$ new states between the states of the reduced MC which is of height at most $|\phi|$, we obtain the claimed bound on the height. ◀

Figure 5 illustrates the transformation of the models as described in the proof sketch. sel_1 and sel_2 are the selections. In the $\mathbf{F}_q, \mathbf{G}_1$-fragment, we directly connected those sets. Here, we insert simple chains between the selections. The construction guarantees that we have at most one state per **F**-formula to satisfy. This is obtained by postponing the satisfaction until the last possible moment before sel.

▶ Remark. Note that the resulting models have a tree shape with BSCCs. There are no non-bottom SCCs. Even if the original model did have such, they are removed by this construction: The reduction algorithm takes care that every non-nested **F**-formula occurs at most once on every path. The inserted chains contain at most one state per nested **F**-formula and do not introduce cycles.

▶ **Example 25.** Consider the formula

$$\phi := \mathbf{F}_{\geq 1/2}(\mathbf{G}_{=1}a) \wedge \mathbf{G}_{=1}(\mathbf{F}_{=1}\neg a \vee \mathbf{F}_{=1}b).$$

Figure 6 (a) shows a model for ϕ. The boxes illustrate the selection of s_0. Figure 6 (b) shows the corresponding reduced chain. However, it is not a model for ϕ. The reason is that neither states satisfying $\neg a$ nor such that satisfy b are reached almost surely from s_0. By including additional states, the chain in Figure 6 (c) corrects this, and thereby we obtain a model of ϕ.

3.6 Finite Models for $\mathbf{F}_q, \mathbf{G}_1, \vee$

In this section, we discuss PCTL($\mathbf{F}, \mathbf{G}$) where **G** appears only with the contraint $q = 1$. Previously for the $\mathbf{F}_q, \mathbf{G}_1$-fragment, i.e. without disjunctions, we started from a model which was unfolded for a number of steps; we simplified such a model by dropping states (including all non-bottom SCCs) and then we inserted simple chains that guarantee the satisfaction of the **G**-nested **F**-formulae. The resulting model thus (i) does not contain any non-bottom SCCs and (ii) the size only depends on the structure of the formula, not on the constraints of the **F**-formulae. However, in the general $\mathbf{F}_q, \mathbf{G}_1, \vee$-fragment, we cannot insert such simple

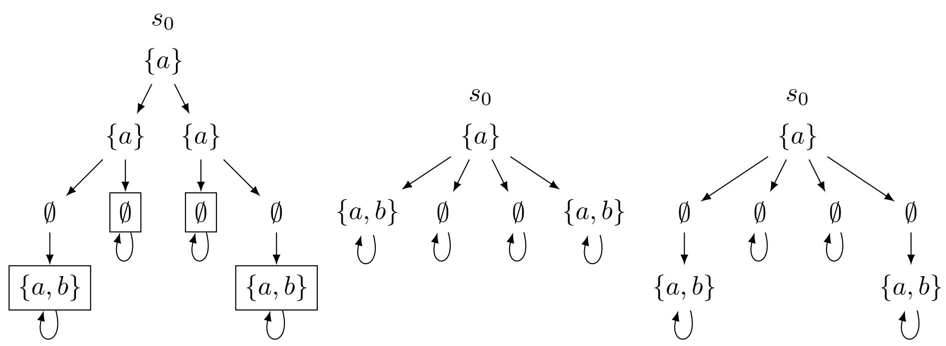

Figure 6 Example of a reduction in the $\mathbf{F_{q/1}, G_1}, \vee$ fragment: (a) original model, (b) reduced model, (c) corrected model.

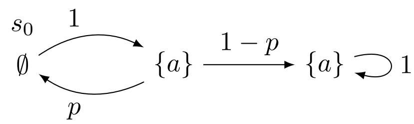

Figure 7 Example model for ϕ.

chains to satisfy nested $\mathbf{F}$s. Instead, we may have to branch at several places. Intuitively, the reason for such complications is the presence of a *repeated, controlled choice*. This enables us to find a formula which requires more complicated models, namely models which either have non-bottom SCCs or are of size which also depends on the constraints and not only on the structure of the formula, or can even be infinite.

▶ **Example 26.** Consider $\phi := \mathbf{G}_{=1}(\mathbf{F}_{=1}(a \wedge \mathbf{F}_{>0}\neg a) \vee a) \wedge \mathbf{F}_{=1}\mathbf{G}_{=1}a \wedge \neg a$. We can try to construct a model for ϕ as follows: Firstly, we have to start at a state which satisfies $\neg a$. This enforces the satisfaction of the first disjunct in the $\mathbf{G}$-formula. Therefore, almost all paths must lead to a state satisfying $a \wedge \mathbf{F}_{>0}\neg a$. This state must eventually reach a state that satisfies $\neg a$ again with positive probability. Hence, we find ourselfs in the same situation as was the case in the initial state. So, we need to either create an SCC of alternating states that satisfy a and $\neg a$, or create an infinite model. If we create an SCC, the side constraint $\mathbf{F}_{=1}\mathbf{G}_{=1}a$ enforces us to eventually leave this SCC. Hence, it is a non-bottom SCC. The MC in Figure 7 is a possible model. From s_0 it models ϕ, for any $p \in (0, 1)$.

Note that the formula given in the example is qualitative. For this fragment, it is known how to solve the satisfiability problem already [7]. However, we can easily adapt the formula to be quantitative. In this case, we might still be able to obtain a model for the quantitative version by keeping its shape and only adapting the probabilties of the model for the qualitative version. The question arises, whether this is possible in general or not. This question remains open and might be interesting for future work.

4 Discussion, Conclusion, and Future Work

We have identified the pattern of the *controlled repeated choice*, i.e. formulae of the form

$$\mathbf{G}_{=1}(\phi_1 \vee \cdots \vee \phi_n)$$

where at least one of the ϕ_i contains an $\mathbf{F}$-formula that has a constraint other than $q = 1$. Additionally, we have "controlling" side constraints, as in Example 26. We have seen that the presence of this pattern enforces more complicated structure of models even in the qualitative setting. This pattern is expressible in the $\mathbf{F}_q, \mathbf{G}_1, \vee$-fragment. Whenever we

(a) drop the side constraints, keeping only the $\mathbf{G}_{=1}$-part, i.e. consider the $\mathbf{G}_1(\mathbf{F}_q, \mathbf{G}_1, \vee)$-fragment, or
(b) drop the disjunction for the choice and consider the $\mathbf{F}_q, \mathbf{G}_1$-fragment, or
(c) drop the quantity of the choice and consider the $\mathbf{F}_{q/1}, \mathbf{G}_1, \vee$-fragment,

the structure is simpler and we obtain decidability. For these fragments, we have even shown that the general satisfiability problem is equivalent to the finite satisfiability problem.

Further, adding quantities to $\mathbf{G}$-constraints obviously also makes the satisfiability problem more complicated. Already for the qualitative $\mathbf{G}_{>0}(\mathbf{F}_{>0})$-fragment, satisfiability and finite satisfiability differ. Nevertheless, we established the decidability of finite satisfiability even for the $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$-fragment.

Consequently, instead of attacking the whole quantitative PCTL or even just PCTL($\mathbf{F}, \mathbf{G}$), we suggest two easier tasks, which should lead to a fundamental increase in understanding the general problem, namely:

- finite (and also general) satisfiability of the $\mathbf{F}_q, \mathbf{G}_1, \vee$-fragment, i.e. PCTL($\mathbf{F}, \mathbf{G}$) where $\mathbf{G}$ is limited to the $= 1$ constraint, and
- infinite satisfiability of the $\mathbf{G}_q(\mathbf{F}_q, \mathbf{G}_q, \vee)$-fragment, i.e. $\mathbf{G}$-formulae of PCTL($\mathbf{F}, \mathbf{G}$).

While the former omits issues stemming from $\mathbf{G}_{>0}$ [7] and only deals with the repeated choice, the latter generalizes the qualitative results for $\mathbf{G}_{>0}, \mathbf{G}_{=1}$ [17, 7] in the presence of general quantitative $\mathbf{F}$'s.

Further, potentially more straight-forward directions include the generalization of the results obtained in this paper to the until- and release-operators instead of future- and globally-operators, respectively, or the introduction of the next-operator.

References

1 C. Baier, M. Größer, M. Leucker, B. Bollig, and F. Ciesinski. Controller synthesis for probabilistic systems. In *Proceedings of IFIP TCS'2004*, pages 493–506. Kluwer, 2004.
2 Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT press, 2008.
3 B. Banieqbal and H. Barringer. Temporal logic with fixed points. In *Temporal Logic in Specification*, volume 398 of *LNCS*, pages 62–74. Springer, 1987.
4 Nathalie Bertrand, John Fearnley, and Sven Schewe. Bounded satisfiability for PCTL. In *Computer Science Logic (CSL'12) - 26th International Workshop/21st Annual Conference of the EACSL, CSL 2012, September 3-6, 2012, Fontainebleau, France*, pages 92–106, 2012.
5 T. Brázdil, V. Brožek, V. Forejt, and A. Kučera. Stochastic games with branching-time winning objectives. In *Proceedings of LICS 2006*, pages 349–358. IEEE, 2006.
6 T. Brázdil, V. Forejt, and A. Kučera. Controller synthesis and verification for Markov decision processes with qualitative branching time objectives. In *Proceedings of ICALP 2008*, LNCS. Springer, 2008.
7 T. Brázdil, V. Forejt, J. Křetínský, and A. Kučera. The satisfiability problem for probabilistic CTL. In *LICS*, pages 391–402. IEEE, 2008. `doi:10.1109/LICS.2008.21`.
8 T. Brázdil, A. Kučera, and O. Stražovský. On the decidability of temporal properties of probabilistic pushdown automata. In Diekert and Durand [11], pages 145–157.
9 Souymodip Chakraborty and Joost-Pieter Katoen. On the satisfiability of some simple probabilistic logics. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 56–65. ACM, 2016.
10 C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *JACM*, 42(4):857–907, 1995.

11 Volker Diekert and Bruno Durand, editors. *STACS 2005, 22nd Annual Symposium on Theoretical Aspects of Computer Science, Stuttgart, Germany, February 24-26, 2005, Proceedings*, volume 3404 of *Lecture Notes in Computer Science*. Springer, 2005.

12 E. A. Emerson and J. Y. Halpern. Decision procedures and expressiveness in the temporal logic of branching time. In *Proceedings of STOC'82*, pages 169–180. ACM, 1982.

13 J. Esparza, A. Kučera, and R. Mayr. Model-checking probabilistic pushdown automata. *Logical Methods in Computer Science*, 2, 2006. doi:10.2168/LMCS-2(1:2)2006.

14 K. Etessami and M. Yannakakis. Recursive Markov chains, stochastic grammars, and monotone systems of non-linear equations. In Diekert and Durand [11], pages 340–352.

15 M. Fischer and R. Ladner. Propositional dynamic logic of regular programs. *JCSS*, 18:194–211, 1979.

16 H. Hansson and B. Jonsson. A logic for reasoning about time and reliability. *FAC*, 6:512–535, 1994.

17 S. Hart and M. Sharir. Probabilistic propositional temporal logics. *Information and Control*, 70(2/3):97–155, 1986.

18 T. Henzinger, O. Kupferman, and R. Majumdar. On the universal and existential fragments of the μ-calculus. *TCS*, 354(2):173–186, 2006.

19 M. Huth and M.Z. Kwiatkowska. Quantitative analysis and model checking. In *Proceedings of LICS'97*, pages 111–122. IEEE, 1997.

20 D. Kozen. A finite-model theorem for the propositional μ-calculus. *Studia Logica*, 47(3):233–241, 1988.

21 S. Kraus and D. J. Lehmann. Decision procedures for time and chance (extended abstract). In *FOCS*, pages 202–209. IEEE, 1983.

22 A. Kučera and O. Stražovský. On the controller synthesis for finite-state Markov decision processes. In *Proceedings of FST&TCS 2005*, volume 3821 of *LNCS*, pages 541–552. Springer, 2005.

23 O. Kupferman and M.Y. Vardi. An automata-theoretic approach to modular model checking. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 22:87–128, 2000.

24 Jan Křetínský and Alexej Rotar. The satisfiability problem for unbounded fragments of probabilistic CTL. Technical report, arXiv.org, 2018. arXiv:1806.11418.

25 D. J. Lehmann and S. Shelah. Reasoning with time and chance (extended abstract). In *ICALP*, volume 154 of *LNCS*, pages 445–457. Springer, 1983.

26 J.R. Norris. *Markov Chains*. Cambridge, 1998.