Qualitative Multi-Objective Reachability for Ordered Branching MDPs

Citation for published version:
Etessami, K & Martinov, E 2020, Qualitative Multi-Objective Reachability for Ordered Branching MDPs. in S Schmitz & I Potapov (eds), *Reachability Problems (RP 2020).* Lecture Notes in Computer Science, vol. 12448, Springer, pp. 67-82, 14th International Conference on Reachability Problems, Paris, France, 19/10/20. https://doi.org/10.1007/978-3-030-61739-4_5

Digital Object Identifier (DOI):
10.1007/978-3-030-61739-4_5

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Reachability Problems (RP 2020)

Qualitative Multi-Objective Reachability for Ordered Branching MDPs*

Kousha Etessami[1] and Emanuel Martinov[1]

School of Informatics, University of Edinburgh, UK
kousha@inf.ed.ac.uk , eo.martinov@gmail.com

Abstract. We study qualitative multi-objective reachability problems for Ordered Branching Markov Decision Processes (OBMDPs), or equivalently context-free MDPs, building on prior results for single-target reachability on Branching Markov Decision Processes (BMDPs).
We provide two separate algorithms for "almost-sure" and "limit-sure" multi-target reachability for OBMDPs. Specifically, given an OBMDP, $\mathcal{A}$, given a starting non-terminal, and given a set of *target* non-terminals K of size $k = |K|$, our first algorithm decides whether the supremum probability, of generating a tree that contains every target non-terminal in set K, is 1. Our second algorithm decides whether there is a strategy for the player to almost-surely (with probability 1) generate a tree that contains every target non-terminal in set K. The two separate algorithms are needed: we give examples showing that indeed "almost-sure" $\neq$ "limit-sure" for multi-target reachability in OBMDPs. Both algorithms run in time $2^{O(k)} \cdot |\mathcal{A}|^{O(1)}$, where $|\mathcal{A}|$ is the bit encoding length of $\mathcal{A}$. Hence they run in P-time when k is fixed, and are fixed-parameter tractable with respect to k. Moreover, we show that the qualitative almost-sure (and limit-sure) multi-target reachability decision problem is in general NP-hard, when k is not fixed.

Keywords: markov decision processes · branching processes · stochastic context-free grammars · multi-objective · reachability · almost-sure · limit-sure

1 Introduction

Ordered Branching Markov Decision Processes (OBMDPs) can be viewed as controlled/probabilistic context-free grammars, but without any terminal symbols, and where moreover the non-terminals are partitioned into two sets: controlled non-terminals and probabilistic non-terminals. Each non-terminal, N, has an associated set of grammar rules of the form $N \to \gamma$, where γ is a (possibly empty) sequence of non-terminals. Each probabilistic non-terminal is equipped with a given probability distribution on its associated grammar rules. For each controlled non-terminal, M, there is an associated non-empty set of available actions, A_M, which is in one-to-one correspondence with the grammar rules of

* A full version [11] of this paper is available at arXiv:2008.10591 .

M. So, for each action, $a \in A_M$, there is an associated grammar rule $M \xrightarrow{a} \gamma$. Given an OBMDP, given a "start" non-terminal, and given a "strategy" for the controller, these together determine a probabilistic process that generates a (possibly infinite) random ordered tree. The tree is formed via the usual parse tree expansion of grammar rules, proceeding generation by generation, in a top-down manner. Starting with a root node labeled by the "start" non-terminal, the ordered tree is generated based on the controller's (possibly randomized) choice of action at each node of the tree that is labeled by a controlled non-terminal, and based on the probabilistic choice of a grammar rule at nodes that are labeled by a probabilistic non-terminal.

We assume that a general *strategy* for the controller can operate as follows: at each node v of the ordered tree, labeled by a controlled non-terminal, the controller (player) can choose its action (or its probability distribution on actions) at v based on the entire "ancestor history" of v, meaning based on the entire sequence of labeled nodes and actions leading from the root node to v, *as well as* based on the ordered position of each of its ancestors (including v itself) among its siblings in the tree.

Ordered Branching Processes (OBPs) are OBMDPs without any controlled non-terminals. Both OBPs and OBMDPs are very similar to classic multi-type branching processes (BPs), and to Branching MDP (BMDPs), respectively. The only difference is that for OB(MD)Ps the generated tree is *ordered*. In particular, the rules for an OBMDP have an ordered *sequence* of non-terminals on their right hand side, whereas there is no such ordering in BPs or BMDPs: each rule for a given type associates an unordered multi-set of "offsprings" of various types to that given type. Branching processes and stochastic context-free grammars have well-known applications in many fields, including in natural language processing, biology/bioinformatics (e.g., [17], population genetics [16], RNA modeling [6], and cancer tumor growth modelling [1, 20]), and physics (e.g., nuclear chain reactions). Generalizing these models to MDPs is natural, and can allow us to study, and to optimize algorithmically, settings where such random processes can partially be controlled.

The single-target reachability objective for OBMDPs amounts to optimizing (maximizing or minimizing) the probability that, starting at a given start (root) non-terminal, the generated tree contains some given target non-terminal. This objective has already been thoroughly studied for BMDPs, as well as for (concurrent) stochastic game generalizations of BMDPs ([9, 10]). Moreover, it turns out that there is really no difference at all between BMDPs and OBMDPs when it comes to the single-target reachability objective: all the algorithmic results from [9, 10] carry over, mutatis mutantis, for OBMDPs, and for their stochastic game generalizations.

A natural generalization of single-target reachability is multi-objective reachability, where the goal is to optimize each of the respective probabilities that the generated tree contains each of several different target non-terminals. (Of course, there may be trade-offs between these different objectives.)

Our main concern in this paper is *qualitative* multi-objective reachability problems, where the aim is to determine whether there is a strategy that guarantees that each of the given set of target non-terminals is almost-surely (respectively, limit-surely) contained in the generated tree, i.e., with probability 1 (respectively, with probability arbitrarily close to 1). In fact, we show that the *almost-sure* and *limit-sure* problems do not coincide. That is, there are OBMDPs for which there is no single strategy that achieves probability exactly 1 for reaching all targets, but where nevertheless, for every $\epsilon > 0$, there is a strategy that guarantees a probability $\geq 1 - \epsilon$, of reaching all targets.

By contrast, for both BMDPs and OBMDPs, for single-target reachability, the *qualitative* almost-sure and limit-sure questions do coincide ([9]).[1]

We give two separate algorithms for almost-sure and limit-sure multi-objective reachability. For the *almost-sure* problem, we are given an OBMDP, a start non-terminal, and a set of target non-terminals, and we must decide whether there exists a strategy using which the process generates, with probability 1, a tree that contains all the given target non-terminals. If the answer is "yes", the algorithm can also construct a (randomized) witness strategy that achieves this.[2] The algorithm for the *limit-sure* problem decides whether the supremum probability of generating a tree that contains all given target non-terminals is 1. If the answer is "yes", the algorithm can also construct, given any $\epsilon > 0$, a randomized non-static strategy that guarantees probability $\geq 1-\epsilon$. The limit-sure algorithm is only slightly more involved.

Both algorithms run in time $2^{O(k)} \cdot |\mathcal{A}|^{O(1)}$, where $|\mathcal{A}|$ is the total bit encoding length of the given OBMDP, $\mathcal{A}$, and $k = |K|$ is the size of the given set K of target non-terminals. Hence they run in polynomial time when k is fixed, and are fixed-parameter tractable with respect to k. Moreover, we show that the qualitative almost-sure (and limit-sure) multi-target reachability decision problem is in general NP-hard, when k is not fixed.

We leave open the decidability of arbitrary boolean combinations of qualitative reachability and non-reachability queries over different target non-terminals. (See the full version for an elaboration on such questions, and algorithms for

[1] The notion of general "strategy" employed for BMDPs in [9] is somewhat different than what we define in this paper for OBMDPs: it allows the controller to not only base its choice at a tree node on the ancestor chain of that node, but on the entire tree up to that "generation". This is needed for BMDPs because there is no ordering available on "siblings" in the tree generated by a BMDP. However, a careful look shows that the results of [9] imply that, for OBMDPs, for single-target reachability, almost-sure and limit-sure reachability also coincide under the notion of "strategy" we have defined in this paper, where choices are based only on the "ancestor history" (with ordering information) of each node in the *ordered* tree. In particular the key "queen/workers" strategy employed for almost-sure (=limit-sure) reachability in [9] can be mimicked using the ordering with respect to siblings that is available in ancestor histories of OBMDPs.

[2] This strategy is, however, necessarily not "static", meaning it must actually use the ancestor history: the action distribution cannot be defined solely based on which non-terminal is being expanded.

some special cases.) Furthermore, we leave open all (both decision and approximation) *quantitative* multi-objective reachability questions, including when the goal is to approximate the tradeoff *pareto curve* of optimal probabilities for different reachability objectives. These are intriguing questions for future research.

Related work. As already mentioned, the single-target reachability problem for OBMDPs (and its stochastic game generalization) is equivalent to the same problem for BMDPs, and was studied in detail in [9, 10], even in the quantitative sense. The same holds for another fundamental objective, namely *termination/extinction*, i.e., where the objective is to optimize the probability that the generated tree is finite. The extinction objective for BMDPs, and the closely related model of 1-exit recursive MDPs, was thoroughly studied in [14, 13, 8], including both qualitative and quantitative algorithmic questions.

Algorithms for checking other properties of BPs and BMDPs have also been investigated before, some of which generalize termination and reachability. In particular, model checking of BPs with properties given by a deterministic parity tree automaton was studied in [3], and in [18] for properties represented by a subclass of alternating parity tree automata. More recently, [19] investigated the determinacy and the complexity of decision problems for ordered branching simple (turn-based) stochastic games with respect to properties defined by finite tree automata defining regular languages on infinite trees. They showed that (unlike the case with reachability) already for some basic regular properties these games are not even determined, meaning they do not have a value. Moreover, they show that for what amounts to OBMDPs with a regular tree objective it is undecidable to compare the optimal probability to a threshold value. Their results do not have implications for (neither quantitative nor qualitative) multi-objective reachability.

Multi-objective reachability and model checking (with respect to omega-regular properties) has been studied for finite-state MDPs in [12], both with respect to qualitative and quantitative problems. In particular, it was shown in [12] that for multi-objective reachability in finite-state MDPs, memoryless (but randomized) strategies are sufficient, that both qualitative and quantitative multi-objective reachability queries can be decided in P-time, and the *Pareto curve* for them can be approximated within a desired error $\epsilon > 0$ in P-time in the size of the MDP and $1/\epsilon$.

Due to space limits, most proofs are omitted (see the full version [11]).

2 Definitions and Background

Rather than providing the most general possible definition of OBMDPs, where rules can have an arbitrarily long string of non-terminals on their right hand side (RHS), to simplify matters, we assume OBMDPs are already in a "simple normal form". This is entirely without loss of generality for our purposes: any OBMDP can be converted efficiently to an "equivalent"[3] one in normal form. This is

[3] Equivalent w.r.t. all (multi-objective) reachability objectives we consider.

directly analogous to standard normal form results for context-free grammars, and to similar prior results established for BMDPs [9].

Definition 1. *An* Ordered Branching Markov Decision Process (OBMDP)*,* $\mathcal{A}$*, (in* simple normal form (SNF)*) is represented by a tuple* $\mathcal{A} = (V, \Sigma, \Gamma, R)$*, where* $V = \{T_1, \ldots, T_n\}$ *is a finite set of non-terminals, and* Σ *is a finite non-empty action alphabet. The set of non-terminals* V *is partitioned into three possible kinds: "controlled" (M-Form) non-terminals, "linear (probabilistic)" (L-Form) non-terminals, and "quadratic (branching)" (Q-Form) non-terminals. For each controlled non-terminal* T_i*,* $\Gamma^i \subseteq \Sigma$ *is a non-empty set of actions available for* T_i*.* R *defines, for each non-terminal* $T_i \in V$*, a set of (probabilistic/controlled)* rules $R(T_i)$*.*

Specifically, the set of rules $R(T_i)$ *associated with non-terminal* $T_i \in V$*, has the following structure, depending on what form (kind) of non-terminal* T_i *is:*

– L-Form*:* T_i *is a "linear" or "probabilistic" non-terminal, the player has no choice of actions, and the associated rules for* T_i *are given by:* $T_i \xrightarrow{p_{i,0}} \varnothing$, $T_i \xrightarrow{p_{i,1}} T_1, \ldots, T_i \xrightarrow{p_{i,n}} T_n$*, where for all* $0 \leq j \leq n$*,* $p_{i,j} \geq 0$ *denotes the probability of each rule, and* $\sum_{j=0}^{n} p_{i,j} = 1$*.*
– Q-Form*:* T_i *is a "quadratic" (or "branching") non-terminal, with a single associated rule (and no associated actions), of the form* $T_i \xrightarrow{1} T_j\ T_{j'}$*.*
– M-Form*:* T_i *a "controlled" non-terminal, with a non-empty set of associated actions* $\Gamma^i = \{a_1, \ldots, a_{m_i}\} \subseteq \Sigma$*, and the associated rules have the form* $T_i \xrightarrow{a_1} T_{j_1}, \ldots, T_i \xrightarrow{a_{m_i}} T_{j_{m_i}}$*.*[4]

We denote by $|\mathcal{A}|$ the total bit encoding length of the OBMDP, where we assume the given rule probabilities are rational numbers represented as usual (with numerator and denominator in binary). If $|\Gamma^i| = 1$ for all controlled non-terminals $T_i \in V$ (meaning the controller has no choices), then the model is an *Ordered Branching Process (OBP)*.

A *derivation* for an OBMDP, starting at some start non-terminal $T_{start} \in V$, is a (possibly infinite) labeled ordered tree, $X = (B, s)$, defined as follows. The set of nodes $B \subseteq \{l, r, u\}^*$ of the tree, X, is a *prefix-closed* subset of $\{l, r, u\}^*$.[5] So each node in B is a string over $\{l, r, u\}$, and if $w = w'a \in B$, where $a \in \{l, r, u\}$, then $w' \in B$. As usual, when $w \in B$ and $w' = wa \in B$, for some $a \in \{l, r, u\}$, we call w the *parent* of w', and we call w' a *child* of w in the tree. A *leaf* of B is a node $w \in B$ that has no children in B. Let $\mathcal{L}_B \subseteq B$ denote the set of all leaves in B. The *root* node is the empty string ε (note that B is prefix-closed, so $\varepsilon \in B$). The function $s : B \to V \cup \{\varnothing\}$ assigns either a non-terminal or the empty symbol as a label to each node of the tree, and must satisfy the following conditions: Firstly, $s(\varepsilon) = T_{start}$, in other words the root must be labeled by the start non-terminal; Inductively, if for any *non-leaf* node $w \in B \setminus \mathcal{L}_B$ we have $s(w) = T_i$, for some $T_i \in V$, then:

[4] We assume, without loss of generality, that for $0 \leq t < t' \leq m_i$, $T_{j_t} \neq T_{j_{t'}}$.
[5] Here 'l', 'r', and 'u', stand for 'left', 'right', and 'unique' child, respectively.

- if T_i is a Q-form (branching) non-terminal, whose associated unique rule is $T_i \xrightarrow{1} T_j\ T_{j'}$, then w must have exactly two children in B, namely $wl \in B$ and $wr \in B$, and moreover we must have $s(wl) = T_j$ and $s(wr) = T_{j'}$.
- if T_i is a L-form (linear/probabilistic) non-terminal, then w must have exactly one child in B, namely wu, and it must be the case that either $s(wu) = T_j$, where there exists some rule $T_i \xrightarrow{p_{i,j}} T_j$ with a positive probability $p_{i,j} > 0$, or else $s(wu) = \varnothing$, where there exists a rule $T_i \xrightarrow{p_{i,0}} \varnothing$, with an empty right hand side, and a positive probability $p_{i,0} > 0$.
- if T_i is a M-form (controlled) non-terminal, then w must have exactly one child in B, namely wu, and it must be the case that $s(wu) = T_{j_t}$, where there exists some rule $T_i \xrightarrow{a_t} T_{j_t}$, associated with some action $a_t \in \Gamma^i$, having non-terminal T_i as its left hand side.

A derivation $X = (B, s)$ is *finite* if the set B is finite. A derivation $X' = (B', s')$ is called a *subderivation* of a derivation $X = (B, s)$, if $B' \subseteq B$ and $s' = s|_{B'}$ (i.e., s' is the function s, restricted to the domain B'). We use $X' \preceq X$ to denote the fact that X' is a subderivation of X.

A *complete* derivation, or a *play*, $X = (B, s)$, is by definition a derivation in which for all leaves $w \in \mathcal{L}_B$, $s(w) = \varnothing$. For a play $X = (B, s)$, and a node $w \in B$, we define the *subplay of X rooted at w*, to be the play $X^w = (B^w, s^w)$, where $B^w = \{w' \in \{l, r, u\}^* \mid ww' \in B\}$ and $s^w : B^w \to V \cup \{\varnothing\}$ is given by, $s^w(w') := s(ww')$ for all $w' \in B^w$.[6] Consider any derivation $X = (B, s)$, and any node $w = w_1 \ldots w_m \in B$, where $w_k \in \{l, r, u\}$ for all $k \in [m]$. We define the *ancestor history* of w to be a sequence $h_w \in V(\{l, r, u\} \times V)^*$, given by $h_w := s(\varepsilon)(w_1, s(w_1))(w_2, s(w_1w_2))(w_3, s(w_1w_2w_3)) \ldots (w_m, s(w_1w_2 \ldots w_m))$. In other words, the ancestor history h_w of node w specifies the sequence of moves that determine each ancestor of w (starting at ε and including w itself), and also specifies the sequence of non-terminals that label each of ancestor of w.

For an OBMDP, $\mathcal{A}$, a sequence $h \in V(\{l, r, u\} \times V)^*$ is called a *valid* ancestor history if there is some derivation $X = (B', s')$ of $\mathcal{A}$, and node $w \in B'$ such that $h = h_w$. We define the *current non-terminal* of such a valid ancestor history h to be $s'(w)$. In other words, it is the non-terminal that labels the last node of the ancestor history h. Let $\texttt{current}(h)$ denote the current non-terminal of h. Let $H_{\mathcal{A}} \subseteq V(\{l, r, u\} \times V)^*$ denote the set of all valid ancestor histories of $\mathcal{A}$. A valid ancestor history $h \in H_{\mathcal{A}}$ is said to *belong to the controller*, if $\texttt{current}(h)$ is a M-form (controlled) non-terminal. Let $H^C_{\mathcal{A}}$ denote the set of all valid ancestor histories of the OBMDP, $\mathcal{A}$, that belong to the controller.

For an OBMDP, $\mathcal{A}$, a *strategy* for the controller is a function, $\sigma : H^C_{\mathcal{A}} \to \Delta(\Sigma)$ from the set of valid ancestor histories belonging to the controller, to probability distributions on actions, such that moreover for any $h \in H^C_{\mathcal{A}}$, if $\texttt{current}(h) = T_i$, then $\sigma(h) \in \Delta(\Gamma^i)$. (In other words, the probability distribution must have

[6] To avoid confusion, note that subderivation and subplay have very different meanings. Saying derivation X is a "subderivation" of X', means that in a sense X is a "prefix" of X', as an ordered tree. Saying play X is a subplay of play X', means X is a "suffix" of X', more specifically X is a subtree rooted at a specific node of X'.

support only on the actions available at the current non-terminal.) Note that the strategy can choose different distributions on actions at different occurrences of the same non-terminal in the derivation tree, even when these occurrences happen to be "siblings" in the tree.

Let Ψ be the set of all strategies. We say $\sigma \in \Psi$ is *deterministic* if for all $h \in H^C_{\mathcal{A}}$, $\sigma(h)$ puts probability 1 on a single action. We say $\sigma \in \Psi$ is *static* if for each M-form (controlled) non-terminal T_i, there is some distribution $\delta_i \in \Delta(\Gamma^i)$, such that for any $h \in H^C_{\mathcal{A}}$ with $\texttt{current}(h) = T_i$, $\sigma(h) = \delta_i$. In other words, a static strategy σ plays exactly the same distribution on actions at every occurrence of each non-terminal T_i, regardless of the ancestor history.

For an OBMDP, $\mathcal{A}$, fixing a start non-terminal T_i, and fixing a strategy σ for the controller, determines a stochastic process that generates a random play, as follows. The process generates a sequence of finite derivations, X_0, X_1, X_2, X_3, …, one for each "generation", such that for all $t \in \mathbb{N}$, $X_t \preceq X_{t+1}$. $X_0 = (B_0, s_0)$ is the initial derivation, at generation 0, and consists of a single (root) node $B_0 = \{\varepsilon\}$, labeled by the start non-terminal, $s_0(\varepsilon) = T_i$. Inductively, for all $t \in \mathbb{N}$ the derivation $X_{t+1} = (B_{t+1}, s_{t+1})$ is obtained from $X_t = (B_t, s_t)$ as follows. For each leaf $w \in \mathcal{L}_{B_t}$:

- if $s_t(w) = T_i$ is a Q-form (branching) non-terminal, whose associated unique rule is $T_i \xrightarrow{1} T_j\ T_{j'}$, then w must have exactly two children in B_{t+1}, namely $wl \in B_{t+1}$ and $wr \in B_{t+1}$, and moreover we must have $s_{t+1}(wl) = T_j$ and $s_{t+1}(wr) = T_{j'}$.
- if $s_t(w) = T_i$ is a L-form (probabilistic) non-terminal, then w has exactly one child in B_{t+1}, namely wu, and for each rule $T_i \xrightarrow{p_{i,j}} T_j$ with $p_{i,j} > 0$, the probability that $s_{t+1}(wu) = T_j$ is $p_{i,j}$, and likewise when $T_i \xrightarrow{p_{i,0}} \varnothing$, is a rule with $p_{i,0} > 0$, then $s_{t+1}(wu) = \varnothing$ with probability $p_{i,0}$.
- if $s_t(w) = T_i$ is a M-form (controlled) non-terminal, then w has exactly one child in B_{t+1}, namely wu, and for each action $a_z \in \Gamma^i$, with probability $\sigma(h_w)(a_z)$, $s_{t+1}(wu) = T_{j_z}$, where $T_i \xrightarrow{a_z} T_{j_z}$ is the rule associated with a_z.

There are no other nodes in B_{t+1}. In particular, if $s_t(w) = \varnothing$, then in B_{t+1} the node w has no children. This defines a stochastic process, $X_0, X_1, X_2, \ldots$, where $X_t \preceq X_{t+1}$, for all $t \in \mathbb{N}$, and such that there is a unique play, $X = \lim_{t\to\infty} X_t$, such that $X_t \preceq X$ for all $t \in \mathbb{N}$. In this sense, the random process defines a probability space of plays.

For our purposes, an *objective* is specified by a property (i.e., a measurable set), $\mathcal{F}$, of plays, whose probability the player wishes to optimize (maximize or minimize). For a property $\mathcal{F}$ and a strategy $\sigma \in \Psi$, let $Pr^{\sigma}_{T_i}[\mathcal{F}]$ denote the probability that starting at non-terminal T_i, under strategy σ, the generated play is in the set $\mathcal{F}$. Let $Pr^{*}_{T_i}[\mathcal{F}] := \sup_{\sigma\in\Psi} Pr^{\sigma}_{T_i}[\mathcal{F}]$. For a non-terminal T_q, $q \in [n]$, let $Reach(T_q)$ denote the set of plays that contain T_q as a label of some node. Let $Reach^{\complement}(T_q)$ denote the complement event, i.e., the set of plays that do <u>not</u> contain T_q. A rather general form of *quantitative* multi-objective reachability decision problems that one might wish to consider is whether there exists a strategy $\sigma' \in \Psi$ such that a boolean combination of statements of the form $Pr^{\sigma'}_{T_i}[\mathcal{F}_j] \triangle_j p_j$

holds, where $\triangle_j \in \{<, \leq, =, \geq, >\}$, and where $\mathcal{F}_j$ is itself a boolean combination (using union and intersection) of (non-)reachability objectives of the form $Reach(T_{j_k})$ and $Reach^C(T_{j_k})$.

Our primary focus is on the following two *qualitative* multi-objective reachability problems. Given an OBMDP, $\mathcal{A}$ with non-terminals $V = \{T_1, \ldots, T_n\}$, given a start non-terminal T_i, and given set $K \subseteq [n]$ of targets, we wish to decide:

- (almost-sure): does there exist $\sigma \in \Psi$ such that $\bigwedge_{q \in K} Pr^{\sigma}_{T_i}[Reach(T_q)] = 1$? (Equivalently, does there exist $\sigma \in \Psi$ s.t. $Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$?[7])
- (limit-sure): Is there, for every $\epsilon > 0$, a $\sigma_\epsilon \in \Psi$, s.t. $\bigwedge_{q \in K} Pr^{\sigma_\epsilon}_{T_i}[Reach(T_q)] \geq 1 - \epsilon$? (Equivalently, is $Pr^{*}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$?[7])

As mentioned, when $|K| = 1$, the almost-sure and limit-sure questions are equivalent ([9]). The following example shows this is not so when $|K| \geq 2$:

Example 1 Consider the OBMDP with non-terminals $\{M, A, R_1, R_2\}$, and with target non-terminals $\{R_1, R_2\}$. M is the only "controlled" non-terminal, and the rules are[8]:

$$M \xrightarrow{a} M\ A \qquad\qquad A \xrightarrow{1/2} R_1$$

$$M \xrightarrow{b} R_2 \qquad\qquad A \xrightarrow{1/2} \varnothing$$

The supremum probability, $Pr^{*}_{M}[Reach(R_1) \cap Reach(R_2)]$, starting with non-terminal M, of reaching both targets is 1. To see this, for any $\epsilon > 0$, let the strategy keep choosing deterministically the action a until $l := \lceil \log_2(\frac{1}{\epsilon}) \rceil$ copies of non-terminal A have been created. Then in the (unique) copy of non-terminal M in generation l the strategy switches deterministically to action b. The probability of reaching target R_2 is 1. The probability of reaching R_1 is $1 - 2^{-l} \geq 1 - \epsilon$. Hence $Pr^{*}_{M}[Reach(R_1) \cap Reach(R_2)] = 1$.

However, $\nexists \sigma \in \Psi :\ \Pr^{\sigma}_{M}[Reach(R_1) \cap Reach(R_2)] = 1$. To see this, note that if the strategy ever puts positive probability on action b in any "round", then with positive probability target R_1 will not be reached in the play. So, to reach target R_1 with probability 1, the strategy must deterministically choose action a forever, from every occurrence of non-terminal M. But if it does this the probability of reaching target R_2 would be 0. $\square$

We now observe (proof in the full version) that qualitative multi-objective reachability problems over an unbounded target set K are in general NP-hard.

Proposition 1.

(1.) The following two problems are both NP-hard: given an OBMDP, a set $K \subseteq [n]$ of target non-terminals, and a start non-terminal $T_i \in V$, decide whether (i) $\exists \sigma \in \Psi : Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$, & (ii) $Pr^{*}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$.

---

[7] The fact that these statements are equivalent is easy to prove; see the full version.

[8] Technically, as given, this OBMDP in not in simple normal form; but this can easily be rectified by using an auxiliary branching non-terminal, Q, adding the rule $Q \xrightarrow{1} M\ A$ and changing the rule $M \xrightarrow{a} M\ A$ to $M \xrightarrow{a} Q$.

(2.) The following problem is coNP-hard: given an OBP (i.e., an OBMDP with no controlled non-terminals, and hence with only one trivial strategy σ), a set $K \subseteq [n]$ of target non-terminals, and a start non-terminal $T_i \in V$, decide whether $Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 0$.

The proof is a reduction from 3-SAT for (1.), and from its complement for (2.).

We shall hereafter often use the notation $T_i \to T_j$ (respectively, $T_i \not\to T_j$), to denote that for non-terminal T_i there exists (respectively, there does *not* exist) either an associated (controlled) rule $T_i \xrightarrow{a} T_j$, where $a \in \Gamma^i$, or an associated probabilistic rule $T_i \xrightarrow{p_{i,j}} T_j$ with positive probability $p_{i,j} > 0$. Similarly let $T_i \to \varnothing$ (respectively, $T_i \not\to \varnothing$), denote that the rule $T_i \xrightarrow{p_{i,0}} \varnothing$ has positive probability $p_{i,0} > 0$ (respectively, has probability $p_{i,0} = 0$).

Definition 2. *The* dependency graph *of an OBMDP, $\mathcal{A}$, is a directed graph that has a node T_i for each non-terminal T_i, and contains an edge (T_i, T_j) if and only if: either $T_i \to T_j$ or there is a rule $T_i \xrightarrow{1} T_j\ T_r$ or a rule $T_i \xrightarrow{1} T_r\ T_j$ in $\mathcal{A}$.*

For an OBMDP, $\mathcal{A}$, with non-terminals set V, we let $G = (U, E)$, with $U = V$, denote the dependency graph of $\mathcal{A}$, and we use $G[C]$ to denote the subgraph of G induced by the subset $C \subseteq U$ of nodes (non-terminals).

Definition 3. *For a directed graph $G = (U, E)$, given a partition of its vertices $U = (U_1, U_P)$, an* end-component *is a set of vertices $C \subseteq U$ such that $G[C]$: (1) is strongly connected; (2) for all $u \in U_P \cap C$ and all $(u, u') \in E$, $u' \in C$; (3) and if $C = \{u\}$ (i.e., $|C| = 1$), then $(u, u) \in E$. A* maximal end-component (MEC) *is an end-component not contained in any larger end-component. A* MEC-decomposition *is a partition of the graph into MECs and nodes that do not belong to any MEC.*

MECs are disjoint and the unique MEC-decomposition of such a directed graph G (with a given partition of its nodes) can be computed in P-time ([5]).[9] More recent work provides more efficient algorithms for computing a MEC-decomposition ([2]). For an OBMDP dependency graph $G = (U, E)$, $U = V$, the partition of U we use is: $U_P := \{T_i \in U \mid T_i \text{ is of } \mathsf{L}\text{-form}\}$ and $U_1 := \{T_i \in U \mid T_i \text{ is of } \mathsf{M}\text{-form or } \mathsf{Q}\text{-form}\}$. We will also be using the notion of a strongly connected component (SCC) of a dependency graph, which can be defined as a MEC where condition (2) from Definition 3 above is not required. As is well-known, an SCC-decomposition of a digraph can be computed in linear time.

3 Algorithm for deciding $Pr^{*}_{T_i}[\bigcap_{q \in K} Reach(T_q)] \stackrel{?}{=} 1$

We first note that there is a (relatively easy) algorithm to compute, for every subset of the target non-terminals $K' \subseteq K$, the sets $Z_{K'} := \{T_i \in V \mid \forall \sigma \in$

[9] In [5], maximal end-components are referred to as *closed components*.

$\Psi: \ Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 0\}$ and $\bar{Z}_{K'} := V - Z_{K'}$. This can be computed, in time via a suitable "attractor set" construction and dynamic programming, using as an initialization step an algorithm from [9, Proposition 4.1] for the single-target case. (See the full version for the algorithm and proof.)

Proposition 2. *Given an OBMDP, $\mathcal{A}$, and a set $K \subseteq [n]$ of $k = |K|$ target non-terminals, there is an algorithm that computes, for every subset of target non-terminals $K' \subseteq K$, the set $Z_{K'} := \{T_i \in V \mid \forall \sigma \in \Psi: \ Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 0\}$. The algorithm runs in time $4^k \cdot |\mathcal{A}|^{O(1)}$. The algorithm can also be augmented to compute a deterministic (non-static) strategy $\sigma'_{K'}$ and a rational value $b_{K'} > 0$, such that for all $T_i \notin Z_{K'}$, $Pr^{\sigma'_{K'}}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] \geq b_{K'} > 0$.*

We now present the algorithm for deciding limit-sure multi-target reachability, i.e., whether $Pr^*_{T_i}[\bigcap_{q \in K} Reach(T_q)] \doteq \sup_{\sigma \in \Psi} Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$.

First, as a preprocessing step, for each subset of target non-terminals $K' \subseteq K$, we compute the set $Z_{K'} := \{T_i \in V \mid \forall \sigma \in \Psi: \ Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 0\}$, using the algorithm from Proposition 2. For every $q \in K$, let AS_q denote the set of non-terminals T_j (including T_q itself) such that $Pr^*_{T_j}[Reach(T_q)] = 1$. The set AS_q can be computed in P-time ([9, Theorem 9.3]), for each target non-terminal T_q, $q \in K$. Moreover, it was proved in [9, Theorem 9.4] that for (O)BMDPs the single-target almost-sure and limit-sure reachability problems coincide. So, for every $q \in K$, there exists a strategy τ_q such that $\forall \ T_j \in AS_q$: $Pr^{\tau_q}_{T_j}[Reach(T_q)] = 1$. Let K'_{-i} denote the set $K' - \{i\}$.

Theorem 1. *The algorithm in Figure 1 computes, given an OBMDP, $\mathcal{A}$, and a set $K \subseteq [n]$ of $k = |K|$ target non-terminals, for each subset $K' \subseteq K$, the set of non-terminals $F_{K'} := \{T_i \in V \mid Pr^*_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 1\}$. The algorithm runs in time $4^k \cdot |\mathcal{A}|^{O(1)}$. Moreover, for each $K' \subseteq K$, given $\epsilon > 0$, the algorithm can also be augmented to compute a randomized non-static strategy $\sigma^{\epsilon}_{K'}$ such that $Pr^{\sigma^{\epsilon}_{K'}}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] \geq 1 - \epsilon$ for all non-terminals $T_i \in F_{K'}$.*

We omit the proof and instead provide some brief intuition for why the algorithm works. (The full proof also describes how the algorithm can be augmented to output, when given $\epsilon > 0$ as input, the witness strategy $\sigma^{\epsilon}_{K'}$.) For any subset $K' \subseteq K$ of target non-terminals, the set $D_{K'}$ contains the non-terminals starting from which, by induction using "smaller" target sets, it immediately follows that limit-sure multi-target reachability of K' holds. For instance, $D_{K'}$ contains any controlled (M-form) non-terminal T_i, $i \in K'$, with a rule $T_i \xrightarrow{a_j} T_j$ such that the remaining targets $K' - \{i\}$ can be limit-surely reached starting from T_j. The set $S_{K'}$ accumulates the non-terminals $T_i \in X = V - (D_{K'} \cup Z_{K'})$ such that there is a value $g > 0$ such that for any $\sigma \in \Psi: \ Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] \leq 1 - g$. In other words, $S_{K'}$ will accumulate those non-terminals starting from which limit-sure reachability definitely does not hold. The loop in step II.5. is an attractor set construction that adds non-terminals T_i to set $S_{K'}$ based on prior membership in $S_{K'}$ of non-terminals appearing on the right-hand side of rules

I. Let $F_{\{q\}} := AS_q$, for each $q \in K$. $F_\emptyset := V$.
II. For $l = 2 \ldots k$:
For every subset of target non-terminals $K' \subseteq K$ of size $|K'| = l$:
1. $D_{K'} := \{T_i \in V - Z_{K'} \mid$ one of the following holds:
   - T_i is of L-form where $i \in K'$, $T_i \not\to \varnothing$ and $\forall T_j \in V$: if $T_i \to T_j$, then $T_j \in F_{K'_{-i}}$.
   - T_i is of M-form where $i \in K'$ and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in F_{K'_{-i}}$.
   - T_i is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ where $i \in K'$ and $\exists K_L \subseteq K'_{-i} : T_j \in F_{K_L} \wedge T_r \in F_{K'_{-i}-K_L}$.
   - T_i is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ where $\exists K_L \subset K'$ $(K_L \neq \emptyset) : T_j \in F_{K_L} \wedge T_r \in F_{K'-K_L}$.$\}$
2. Repeat until no change has occurred to $D_{K'}$:
   (a) add $T_i \notin D_{K'}$ to $D_{K'}$, if of L-form, $T_i \not\to \varnothing$ and $\forall T_j \in V$: if $T_i \to T_j$, then $T_j \in D_{K'}$.
   (b) add $T_i \notin D_{K'}$ to $D_{K'}$, if of M-form and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in D_{K'}$.
   (c) add $T_i \notin D_{K'}$ to $D_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in D_{K'} \vee T_r \in D_{K'}$.
3. Let $X := V - (D_{K'} \cup Z_{K'})$.
4. Initialize $S_{K'} := \{T_i \in X \mid$ either $i \in K'$, or T_i is of L-form and $T_i \to \varnothing \vee T_i \to T_j,\ T_j \in Z_{K'}\} \cup \bigcup_{\emptyset \subset K'' \subset K'} (X \cap S_{K''})$.
5. Repeat until no change has occurred to $S_{K'}$:
   (a) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of L-form and $T_i \to T_j,\ T_j \in S_{K'} \cup Z_{K'}$.
   (b) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of M-form and $\forall a \in \Gamma^i :\ T_i \xrightarrow{a} T_j,\ T_j \in S_{K'} \cup Z_{K'}$.
   (c) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in S_{K'} \cup Z_{K'} \wedge T_r \in S_{K'} \cup Z_{K'}$.
6. $\mathcal{C} \leftarrow$ MEC decomposition of $G[X - S_{K'}]$.
7. For every $q \in K'$, let $H_q := \{T_i \in X - S_{K'} \mid T_i$ is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $((T_j \in X - S_{K'} \wedge T_r \in \bar{Z}_{\{q\}}) \vee (T_j \in \bar{Z}_{\{q\}} \wedge T_r \in X - S_{K'}))\}$.
8. Let $F_{K'} := \bigcup\{C \in \mathcal{C} \mid P_C = K' \vee (P_C \neq \emptyset \wedge P_C \neq K' \wedge \exists T_i \in C, \exists a \in \Gamma^i : T_i \xrightarrow{a} T_j,\ T_j \in F_{K'-P_C})\}$, where $P_C = \{q \in K' \mid C \cap H_q \neq \emptyset\}$.
9. Repeat until no change has occurred to $F_{K'}$:
   (a) add $T_i \in X-(S_{K'} \cup F_{K'})$ to $F_{K'}$, if of L-form and $T_i \to T_j,\ T_j \in F_{K'} \cup D_{K'}$.
   (b) add $T_i \in X - (S_{K'} \cup F_{K'})$ to $F_{K'}$, if of M-form and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in F_{K'}$.
   (c) add $T_i \in X - (S_{K'} \cup F_{K'})$ to $F_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in F_{K'} \vee T_r \in F_{K'}$.
10. If $X \neq S_{K'} \cup F_{K'}$, let $S_{K'} := X - F_{K'}$ and go to step 5.
11. Else, i.e., if $X = S_{K'} \cup F_{K'}$, let $F_{K'} := F_{K'} \cup D_{K'}$.

III. **Output** F_K.

Fig. 1. Algorithm for limit-sure multi-target reachability. The output is the set $F_K = \{T_i \in V \mid Pr^*_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1\}$.

for non-terminal T_i. Step II.6. then builds a MEC-decomposition of the dependency graph $G[X - S_{K'}]$ induced by the remaining non-terminals in set $X - S_{K'}$; step II.8. identifies those MECs, C, where starting at a non-terminal in C the following is observed: the branching (Q-Form) non-terminals in C spawn two children each, at least one of which belongs to C, and other spawned children of the branching non-terminals in C can collectively reach a non-empty subset P_C of (or in the best case, all of) the target set K' with a positive probability (bounded away from zero); the player can choose to delay arbitrarily long the moment to select an action that "exits" C and, thus, can choose to reach the targets in set P_C with probability arbitrarily close to 1; and once the player chooses to "exit" C, it does so in a non-terminal that can limit-surely reach the rest of the targets in set $K' - P_C$. Step II.9. accumulates in the set $F_{K'}$ the set of non-terminals that can almost-surely reach the set $D_{K'}$ or one of the MECs computed in step II.8. A key assertion is this: if in step II.11. we find all non-terminals from the set X are already either in set $S_{K'}$ or in set $F_{K'}$, then we are done: $F_{K'} \cup D_{K'}$ must constitute the set of all non-terminals starting in which the player can force limit-sure reachability of all targets in set K' in the same play; otherwise, all non-terminals in set $X - (F_{K'} \cup S_{K'})$ can be added to set $S_{K'}$, meaning that starting at any of these non-terminals, limit-sure reachability of all target non-terminals in set K' cannot be achieved. This latter assertion is not obvious, but it is true (see the proof in the full version).

4 Algorithm for deciding whether $\exists \sigma \in \Psi$: $Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1$

We now present the algorithm (Figure 2) for deciding almost-sure multi-target reachability for a given OBMDP, $\mathcal{A}$, i.e., given a set $K \subseteq [n]$ of $k = |K|$ target non-terminals and a starting non-terminal T_i, deciding whether there is a strategy for the player under which the probability of generating a play that contains all target non-terminals from set K is 1. Again, as in the limit-sure algorithm, for each subset of target non-terminals $K' \subseteq K$, as a preprocessing step we compute the set $Z_{K'} := \{T_i \in V \mid \forall \sigma \in \Psi : \; Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 0\}$. And for every $q \in K$, we compute (in P-time) the set AS_q of non-terminals T_j (including target T_q itself) such that there is a strategy τ with $Pr^{\tau}_{T_j}[Reach(T_q)] = 1$.

Theorem 2. *The algorithm in Figure 2 computes, given an OBMDP, $\mathcal{A}$, and a set $K \subseteq [n]$ of $k = |K|$ target non-terminals, for each subset $K' \subseteq K$, the set of non-terminals $F_{K'} := \{T_i \in V \mid \exists \sigma \in \Psi : \; Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 1\}$. The algorithm runs in time $4^k \cdot |\mathcal{A}|^{O(1)}$. Moreover, for each $K' \subseteq K$, the algorithm can also be augmented to compute a randomized non-static strategy $\sigma^*_{K'}$ such that $Pr^{\sigma^*_{K'}}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] = 1$ for all non-terminals $T_i \in F_{K'}$.*

We again omit the proof and instead provide a brief sketch of why the algorithm works. (The full proof also describes how the algorithm can be augmented to output the witness strategy $\sigma^*_{K'}$.) Both the sketch and the algorithm itself

I. Let $F_{\{q\}} := AS_q$, for each $q \in K$. $F_\emptyset := V$.
II. For $l = 2 \ldots k$:
For every subset of target non-terminals $K' \subseteq K$ of size $|K'| = l$:
1. $D_{K'} := \{T_i \in V - Z_{K'} \mid$ one of the following holds:
   - T_i is of L-form where $i \in K'$, $T_i \not\to \varnothing$ and $\forall T_j \in V$: if $T_i \to T_j$, then $T_j \in F_{K'_{-i}}$.
   - T_i is of M-form where $i \in K'$ and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in F_{K'_{-i}}$.
   - T_i is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ where $i \in K'$ and $\exists K_L \subseteq K'_{-i} : T_j \in F_{K_L} \wedge T_r \in F_{K'_{-i} - K_L}$.
   - T_i is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ where $\exists K_L \subset K'$ $(K_L \neq \emptyset) : T_j \in F_{K_L} \wedge T_r \in F_{K' - K_L}$.$\}$
2. Repeat until no change has occurred to $D_{K'}$:
   (a) add $T_i \notin D_{K'}$ to $D_{K'}$, if of L-form, $T_i \not\to \varnothing$ and $\forall T_j \in V$: if $T_i \to T_j$, then $T_j \in D_{K'}$.
   (b) add $T_i \notin D_{K'}$ to $D_{K'}$, if of M-form and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in D_{K'}$.
   (c) add $T_i \notin D_{K'}$ to $D_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in D_{K'} \vee T_r \in D_{K'}$.
3. Let $X := V - (D_{K'} \cup Z_{K'})$.
4. Initialize $S_{K'} := \{T_i \in X \mid$ either $i \in K'$, or T_i is of L-form and $T_i \to \varnothing \vee T_i \to T_j,\ T_j \in Z_{K'}\} \ \cup\ \bigcup_{\emptyset \subset K'' \subset K'} (X \cap S_{K''})$.
5. Repeat until no change has occurred to $S_{K'}$:
   (a) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of L-form and $T_i \to T_j,\ T_j \in S_{K'} \cup Z_{K'}$.
   (b) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of M-form and $\forall a \in \Gamma^i :\ T_i \xrightarrow{a} T_j,\ T_j \in S_{K'} \cup Z_{K'}$.
   (c) add $T_i \in X - S_{K'}$ to $S_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in S_{K'} \cup Z_{K'} \wedge T_r \in S_{K'} \cup Z_{K'}$.
6. $\mathcal{C} \leftarrow$ SCC decomposition of $G[X - S_{K'}]$.
7. For every $q \in K'$, let $H_q := \{T_i \in X - S_{K'} \mid T_i$ is of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $((T_j \in X - S_{K'} \wedge T_r \in \bar{Z}_{\{q\}}) \vee (T_j \in \bar{Z}_{\{q\}} \wedge T_r \in X - S_{K'}))\}$.
8. Let $F_{K'} := \bigcup \{\cup_{q \in K'} (H_q \cap C) \mid C \in \mathcal{C}$ s.t. $\forall q' \in K' : H_{q'} \cap C \neq \emptyset\}$.
9. Repeat until no change has occurred to $F_{K'}$:
   (a) add $T_i \in X - (S_{K'} \cup F_{K'})$ to $F_{K'}$, if of L-form and $T_i \to T_j,\ T_j \in F_{K'} \cup D_{K'}$.
   (b) add $T_i \in X - (S_{K'} \cup F_{K'})$ to $F_{K'}$, if of M-form and $\exists a^* \in \Gamma^i : T_i \xrightarrow{a^*} T_j,\ T_j \in F_{K'}$.
   (c) add $T_i \in X - (S_{K'} \cup F_{K'})$ to $F_{K'}$, if of Q-form $(T_i \xrightarrow{1} T_j\ T_r)$ and $T_j \in F_{K'} \vee T_r \in F_{K'}$.
10. If $X \neq S_{K'} \cup F_{K'}$, let $S_{K'} := X - F_{K'}$ and go to step 5.
11. Else, i.e., if $X = S_{K'} \cup F_{K'}$, let $F_{K'} := F_{K'} \cup D_{K'}$.

III. **Output** F_K.

Fig. 2. Algorithm for almost-sure multi-target reachability. The output is the set $F_K = \{T_i \in V \mid \exists \sigma \in \Psi :\ Pr^{\sigma}_{T_i}[\bigcap_{q \in K} Reach(T_q)] = 1\}$.

are very similar to that of the limit-sure case, but differ in some crucial details. Not only do the two algorithms differ in steps II.6. and II.8., but moreover the interpretation of various sets being accumulated in the two algorithms changes (in order to correspond to the appropriate meaning in the context of almost-sure reachability). For any subset $K' \subseteq K$ of target non-terminals, the set $D_{K'}$ contains the non-terminals starting from which, by induction using "smaller" target sets, it immediately follows that almost-sure multi-target reachability is satisfied. The set $S_{K'}$ accumulates the non-terminals $T_i \in X = V - (D_{K'} \cup Z_{K'})$ such that $\forall \sigma \in \Psi : \; Pr^{\sigma}_{T_i}[\bigcap_{q \in K'} Reach(T_q)] < 1$. In other words, $S_{K'}$ will accumulate those non-terminals starting from which almost-sure reachability definitely does not hold. The loop in step II.5. is again an attractor set construction that adds non-terminals T_i to set $S_{K'}$ based on prior membership in $S_{K'}$ of non-terminals appearing on the right-hand side of rules for non-terminal T_i. Step II.6. then builds a SCC-decomposition of the dependency graph $G[X - S_{K'}]$ induced by the remaining non-terminals in set $X - S_{K'}$; step II.8. identifies those branching (Q-form) non-terminals that belong to SCCs, C, where the following is true for each such C: the Q-form non-terminals in C (that have been identified in step II.8.) spawn two children each, at least one of which belongs to C, and the other spawned children of these same branching non-terminals can collectively reach all the targets in set K' with a positive probability (bounded away from zero).[10] Step II.9. accumulates in the set $F_{K'}$ the set of non-terminals that can almost-surely reach the set $D_{K'}$ or the Q-form non-terminals computed in step II.8. A key assertion is this: if in step II.11. we find all non-terminals from the set X are already either in set $S_{K'}$ or in set $F_{K'}$, then we are done: $F_{K'} \cup D_{K'}$ must constitute the set of all non-terminals starting in which the player can force almost-sure reachability of all targets in set K' in the play[11]; otherwise, all non-terminals in set $X - (F_{K'} \cup S_{K'})$ can be added to set $S_{K'}$, meaning that starting at any of these non-terminals, almost-sure reachability of all target non-terminals in set K' cannot be achieved. The reason why this last assertion holds is again not obvious, but it is true (see the proof in the full version).

[10] Note that this differs crucially from the situation in the limit-sure algorithm, where the other spawned children of these branching nodes in C were only able to reach a non-empty *subset* P_C of K' with a positive probability (bounded away from zero), not necessarily the entire set K'.

[11] A helpful observation here is this: in the limit-sure algorithm we were identifying MECs, where the choice of when to "exit" the MEC is entirely controlled by the player. In the almost-sure algorithm we instead identify SCCs. Even though there may also be purely probabilistic (i.e., not controlled) opportunities of "exiting" such a "good" SCC, C (specifically, an SCC C that is identified and used in step II.8.), due to the way the set $F_{K'} = X - S_{K'}$ is constructed (and due to properties of its associated witness strategy $\sigma^*_{K'}$, which is described in the full proof), we can show that even when C is "exited" we still stay inside the set $F_{K'}$, and eventually hit a SCC, C', which can only be "exited" probabilistically to $D_{K'}$, and where, moreover, for each target in K' there is a branching (Q-Form) node in the SCC, C', whose "extra" child can hit that target with positive probability (bounded away from zero).

References

1. Bozic, et. al. Evolutionary dynamics of cancer in response to targeted combination therapy. *eLife*, volume 2, pages e00747, 2013.
2. K. Chatterjee and M. Henzinger. Efficient and Dynamic Algorithms for Alternating Büchi Games and Maximal End-Component Decomposition. *Journal of the ACM*, 61(3):15:1–15:40, 2014.
3. T. Chen, K. Dräger, and S. Kiefer: Model Checking Stochastic Branching Processes. In Proc. of *MFCS'12*, Springer LNCS 7464, pages 271–282, 2012.
4. A. Condon. The complexity of stochastic games. *Inf. & Comp.*, 96(2):203–224, 1992.
5. C. Courcoubetis and M. Yannakakis. Markov decision processes and regular events. *IEEE Trans. on Automatic Control*, 43(10):1399–1418, 1998.
6. R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. *Cambridge University Press*, 1998.
7. K. Etessami, A. Stewart, and M. Yannakakis. A Polynomial-time algorithm for Computing Extinction Probabilities of Multi-type Branching Processes. *SIAM J. on Computing*, 46(5):1515–1553, 2017. (Incorporates part of the work of a conference paper in STOC'12.)
8. K. Etessami, A. Stewart, and M. Yannakakis. Polynomial-time algorithms for Branching Markov Decision Processes, and Probabilistic Min(Max) Polynomial Bellman Equations. *Math. Oper. Res.*, 45(1):34–62,2020. (Conference version in ICALP'12.)
9. K. Etessami, A. Stewart, and M. Yannakakis. Greatest Fixed Points of Probabilistic Min/Max Polynomial Equations, and Reachability for Branching Markov Decision Processes. *Inf. & Comp.*, 261(2):355–382, 2018. (Conference version in ICALP'15.)
10. K. Etessami, E. Martinov, A. Stewart, and M. Yannakakis. Reachability for Branching Concurrent Stochastic Games. In *Proc. 46th Int. Coll. on Automata, Languages and Programming (ICALP)*, 2019. (All references are to the full preprint arXiv:1806.03907 .)
11. K. Etessami and E. Martinov. Qualitative Multi-Objective Reachability for Ordered Branching MDPs. Full preprint arXiv:2008.10591, 2020.
12. K. Etessami, M. Kwiatkowska, M. Y. Vardi, and M. Yannakakis. Multi-Objective Model Checking of Markov Decision Processes. *Logical Methods in Computer Science* 4(4), 2008.
13. K. Etessami and M. Yannakakis. Recursive Concurrent Stochastic Games. *Logical Methods in Computer Science*, volume 4(4), 2008.
14. K. Etessami and M. Yannakakis. Recursive Markov Decision Processes and Recursive Stochastic Games. *Journal of the ACM*, 62(2):1–69, 2015.
15. K. Etessami and M. Yannakakis. Recursive Markov Chains, Stochastic Grammars, and Monotone Systems of Nonlinear Equations. *Journal of the ACM*, 56(1), 2009.
16. P. Haccou, P. Jagers, and V. A. Vatutin. Branching Processes: Variation, Growth, and Extinction of Populations. *Cambridge University Press*, 2005.
17. M. Kimmel and D. E. Axelrod. Branching processes in biology. *Springer*, 2002.
18. H. Michalewski and M. Mio. On the Problem of Computing the Probability of Regular Sets of Trees. In *Proc. of FSTTCS'15*, pp. 489–502, 2015.
19. M. Przybyłko and M. Skrzypczak. On the Complexity of Branching Games with Regular Conditions. In *Proc. of MFCS'16*, LIPICS, volume 78, 2016.

20. G. Reiter, I. Bozic, K. Chatterjee, and M. A. Nowak. TTP: Tool for tumor progression. In *Proc. of CAV'2013*, pp. 101–106, Springer LNCS 8044, 2013.
21. A. P. Sistla and E. M. Clarke. The Complexity of Propositional Linear Temporal Logics. *J. of ACM*, 32(3):733–749, 1985.