Theoretical Computer Science 573 (2015) 71–89

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Average case analysis of the classical algorithm for Markov decision processes with Büchi objectives ☆

Krishnendu Chatterjee [a], Manas Joglekar [b], Nisarg Shah [c,*]

[a] IST Austria, Austria
[b] Stanford University, USA
[c] Carnegie Mellon University, USA

ARTICLE INFO

Article history:
Received 5 July 2013
Received in revised form 14 October 2014
Accepted 30 January 2015
Available online 7 February 2015
Communicated by Z. Esik

Keywords:
Average-case analysis
Büchi objectives
Markov decision processes (MDPs)
Random graphs

ABSTRACT

We consider Markov decision processes (MDPs) with specifications given as Büchi (liveness) objectives, and examine the problem of computing the set of *almost-sure* winning vertices such that the objective can be ensured with probability 1 from these vertices. We study for the first time the average-case complexity of the classical algorithm for computing the set of almost-sure winning vertices for MDPs with Büchi objectives. Our contributions are as follows: First, we show that for MDPs with constant out-degree the expected number of iterations is at most logarithmic and the average-case running time is linear (as compared to the worst-case linear number of iterations and quadratic time complexity). Second, for the average-case analysis over all MDPs we show that the expected number of iterations is constant and the average-case running time is linear (again as compared to the worst-case linear number of iterations and quadratic time complexity). Finally we also show that when all MDPs are equally likely, the probability that the classical algorithm requires more than a constant number of iterations is exponentially small.

1. Introduction

In this work, we consider the qualitative analysis of Markov decision processes with Büchi (liveness) objectives, and establish optimal bounds for the average case complexity. We start by briefly describing the model and the objectives, then the significance of qualitative analysis, followed by the previous results, and finally our contributions.

Markov decision processes. *Markov decision processes (MDPs)* are standard models for probabilistic systems that exhibit both probabilistic and nondeterministic behavior [19], and widely used in verification of probabilistic systems [1,26]. MDPs have been used to model and solve control problems for stochastic systems [18]: there, nondeterminism represents the freedom of the controller to choose a control action, while the probabilistic component of the behavior describes the system response to control actions. MDPs have also been adopted as models for concurrent probabilistic systems [14], probabilistic systems

☆ A preliminary version appeared in the proceedings of 32nd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS), 2012. The research was supported by FWF Grant No. P 23499-N23, FWF NFN Grant No. S11407-N23 (RiSE), ERC Start Grant (279307: Graph Games), and the Microsoft Faculty Fellows Award. Nisarg Shah is also supported by NSF Grant CCF-1215883.

* Corresponding author.
E-mail addresses: Krishnendu.Chatterjee@ist.ac.at (K. Chatterjee), manasrj@stanford.edu (M. Joglekar), nkshah@cs.cmu.edu (N. Shah).

http://dx.doi.org/10.1016/j.tcs.2015.01.050
0304-3975/

operating in open environments [23], under-specified probabilistic systems [2], and applied in diverse domains [26]. A *specification* describes the set of desired behaviors of the system, which in the verification and control of stochastic systems is typically an ω-regular set of paths. The class of ω-regular languages extends classical regular languages to infinite strings, and provides a robust specification language to express all commonly used specifications, such as safety, liveness, fairness, etc. [25]. Parity objectives are a canonical way to define such ω-regular specifications. Thus MDPs with parity objectives provide the theoretical framework to study problems such as the verification and control of stochastic systems.

Qualitative and quantitative analysis. The analysis of MDPs with parity objectives can be classified into qualitative and quantitative analysis. Given an MDP with parity objective, the *qualitative analysis* asks for the computation of the set of vertices from where the parity objective can be ensured with probability 1 (almost-sure winning). The more general *quantitative analysis* asks for the computation of the maximal (or minimal) probability at each state with which the controller can satisfy the parity objective.

Importance of qualitative analysis. The qualitative analysis of MDPs is an important problem in verification that is of interest independent of the quantitative analysis problem. There are many applications where we need to know whether the correct behavior arises with probability 1. For instance, when analyzing a randomized embedded scheduler, we are interested in whether every thread progresses with probability 1 [5]. Even in settings where it suffices to satisfy certain specifications with probability $p < 1$, the correct choice of p is a challenging problem, due to the simplifications introduced during modeling. For example, in the analysis of randomized distributed algorithms it is quite common to require correctness with probability 1 (see, e.g., [21,20,24]). Furthermore, in contrast to quantitative analysis, qualitative analysis is robust to numerical perturbations and modeling errors in the transition probabilities, and consequently the algorithms for qualitative analysis are combinatorial. Finally, for MDPs with parity objectives, the best known algorithms and all algorithms used in practice first perform the qualitative analysis, and then perform a quantitative analysis on the result of the qualitative analysis [14,15,3,4,6,12]. Thus qualitative analysis for MDPs with parity objectives is one of the most fundamental and core problems in verification of probabilistic systems.

Previous results. The qualitative analysis for MDPs with parity objectives is achieved by iteratively applying solutions of the qualitative analysis of MDPs with Büchi objectives [14,15,12]. The qualitative analysis of an MDP with a parity objective with d priorities can be achieved by $O(d)$ calls to an algorithm for qualitative analysis of MDPs with Büchi objectives, and hence we focus on MDPs with Büchi objectives. The qualitative analysis problem for MDPs with Büchi objectives has been widely studied. The classical algorithm for the problem was given in [14,15], and the worst case running time of the classical algorithm is $O(n \cdot m)$ time, where n is the number of vertices, and m is the number of edges of the MDP. Many improved algorithms have also been given in the literature, such as [11,7–10], and several special cases have also been studied [13], and the current best known worst case complexity of the problem is $O(\min\{n^2, m \cdot \sqrt{m}\})$. Moreover, there exists a family of MDPs where the running time of the improved algorithms match the above bound. While the worst case complexity of the problem has been studied, to the best of our knowledge the average case complexity of none of the algorithms has been studied in the literature.

Our contribution. In this work we study for the first time the average case complexity of the qualitative analysis of MDPs with Büchi objectives. Specifically we study the average case complexity of the classical algorithm for the following two reasons: First, the classical algorithm is very simple and appealing as it iteratively uses solutions of the standard graph reachability and alternating graph reachability algorithms, and can be implemented efficiently by symbolic algorithms. Second, while more involved algorithms that improve the worst case complexity have been proposed [11,7–10], it has also been established in [8,10] that there are simple variants of the involved algorithms that require at most a linear running time in addition to the time of the classical algorithm, and hence the average case complexity of these variants is no more than the average case complexity of the classical algorithm. We study the average case complexity of the classical algorithm and establish that compared to the quadratic worst case complexity, the average case complexity is linear. Our main contributions are summarized below:

1. *MDPs with constant out-degree.* We first consider MDPs with constant out-degree. In practice, MDPs often have constant out-degree: for example, see [16] for MDPs with large state space but constant number of actions, or [18,22] for examples from inventory management where MDPs have constant number of actions (the number of actions correspond to the out-degree of MDPs). We consider MDPs where the out-degree of every vertex is fixed and given. The out-degree of a vertex v is d_v and there are constants $d_{\min}$ and $d_{\max}$ such that for every v we have $d_{\min} \le d_v \le d_{\max}$. Moreover, every subset of the set of vertices of size d_v is equally likely to be the neighbor set of v, independent of the neighbor sets of other vertices. We show that the expected number of iterations of the classical algorithm is at most logarithmic ($O(\log n)$), and the average case running time is linear ($O(n)$) (as compared to the worst case linear number of iterations and quadratic $O(n^2)$ time complexity of the classical algorithm, and the current best known $O(n \cdot \sqrt{n})$ worst case complexity). The average case complexity of this model implies the same average case complexity for several related models of MDPs with constant out-degree. For further discussion on this, see Remark 2.

2. *MDPs in the Erdös–Rényi model.* To consider the average case complexity over all MDPs, we consider MDPs where the underlying graph is a random directed graph according to the classical Erdös–Rényi random graph model [17]. We consider random graphs $\mathcal{G}_{n,p}$, over n vertices where each edge exists with probability p (independently of other edges). To analyze the average case complexity over all MDPs with all graphs equally likely, we need to consider the $\mathcal{G}_{n,p}$ model with $p = \frac{1}{2}$ (i.e., each edge is present or absent with equal probability, and thus all graphs are considered equally likely). We show a stronger result (than only $p = \frac{1}{2}$) that if $p \geq \frac{c \cdot \log(n)}{n}$, for some constant $c > 2$, then the expected number of iterations of the classical algorithm is constant ($O(1)$), and the average case running time is linear (again as compared to the worst case linear number of iterations and quadratic time complexity). Note that we obtain that the average case (when $p = \frac{1}{2}$) running time for the classical algorithm is linear over all MDPs (with all graphs equally likely) as a special case of our results for $p \geq \frac{c \cdot \log(n)}{n}$, for any constant $c > 2$, since $\frac{1}{2} \geq \frac{3 \cdot \log(n)}{n}$ for $n \geq 17$. Moreover we show that when $p = \frac{1}{2}$ (i.e., all graphs are equally likely), the probability that the classical algorithm will require more than constantly many iterations is exponentially small in n (less than $\left(\frac{3}{4}\right)^n$).

Implications of our results. We now discuss several implications of our results. First, since we show that the classical algorithm has average case linear time complexity, it follows that the average case complexity of qualitative analysis of MDPs with Büchi objectives is linear time. Second, since qualitative analysis of MDPs with Büchi objectives is a more general problem than reachability in graphs (graphs are a special case of MDPs and reachability objectives are a special case of Büchi objectives), the best average case complexity that can be achieved is linear. Hence our results for the average case complexity are tight. Finally, since for the improved algorithms there are simple variants that never require more than linear time as compared to the classical algorithm it follows that the improved algorithms also have average case linear time complexity. Thus we complete the average case analysis of the algorithms for the qualitative analysis of MDPs with Büchi objectives. In summary our results show that the classical algorithm (the most simple and appealing algorithm) has excellent and optimal (linear-time) average case complexity as compared to the quadratic worst case complexity.

Technical contributions. The two key technical difficulties to establish our results are as follows: (1) Though there are many results for random undirected graphs, for the average case analysis of the classical algorithm we need to analyze random directed graphs; and (2) in contrast to other results related to random undirected graphs that prove results for almost all vertices, the classical algorithm stops only when all vertices satisfy a certain reachability property; and hence we need to prove results for all vertices (as compared to almost all vertices). In this work we set up novel recurrence relations to estimate the expected number of iterations, and the average case running time of the classical algorithm. Our key technical results prove many interesting inequalities related to the recurrence relation for reachability properties of random directed graphs to establish the desired result. We believe the new interesting results related to reachability properties we establish for random directed graphs will find future applications in average case analysis of other algorithms related to verification.

2. Definitions

Markov decision processes (MDPs). A *Markov decision process (MDP)* $G = ((V, E), (V_1, V_P), \delta)$ consists of a directed graph (V, E), a partition (V_1, V_P) of the *finite* set V of vertices, and a probabilistic transition function $\delta \colon V_P \to \mathcal{D}(V)$, where $\mathcal{D}(V)$ denotes the set of probability distributions over the vertex set V. The vertices in V_1 are the *player*-1 vertices, where player 1 decides the successor vertex, and the vertices in V_P are the *probabilistic (or random)* vertices, where the successor vertex is chosen according to the probabilistic transition function δ. We assume that for $u \in V_P$ and $v \in V$, we have $(u, v) \in E$ iff $\delta(u)(v) > 0$, and we often write $\delta(u, v)$ for $\delta(u)(v)$. For a vertex $v \in V$, we write $E(v)$ to denote the set $\{u \in V \mid (v, u) \in E\}$ of possible out-neighbors, and $|E(v)|$ is the out-degree of v. For technical convenience we assume that every vertex in the graph (V, E) has at least one outgoing edge, i.e., $E(v) \neq \emptyset$ for all $v \in V$.

Plays, strategies and probability measure. An infinite path, or a *play*, of the graph G is an infinite sequence $\omega = \langle v_0, v_1, v_2, \ldots \rangle$ of vertices such that $(v_k, v_{k+1}) \in E$ for all $k \in \mathbb{N}$. We write Ω for the set of all plays, and for a vertex $v \in V$, we write $\Omega_v \subseteq \Omega$ for the set of plays that start from the vertex v. A *strategy* for player 1 is a function $\sigma \colon V^* \cdot V_1 \to \mathcal{D}(V)$ that chooses the probability distribution over the successor vertices for all finite sequences $\vec{w} \in V^* \cdot V_1$ of vertices ending in a player-1 vertex (the sequence represents a prefix of a play). A strategy must respect the edge relation: for all $\vec{w} \in V^*$ and $u \in V_1$, if $\sigma(\vec{w} \cdot u)(v) > 0$, then $v \in E(u)$. Let Σ denote the set of all strategies. Once a starting vertex $v \in V$ and a strategy $\sigma \in \Sigma$ is fixed, the outcome of the MDP is a random walk ω_v^σ for which the probabilities of events are uniquely defined, where an *event* $\mathcal{A} \subseteq \Omega$ is a measurable set of plays. For a vertex $v \in V$ and an event $\mathcal{A} \subseteq \Omega$, we write $\mathbb{P}_v^\sigma(\mathcal{A})$ for the probability that a play belongs to $\mathcal{A}$ if the game starts from the vertex v and player 1 follows the strategy σ.

Objectives. We specify *objectives* for the player 1 by providing a set of *winning* plays $\Phi \subseteq \Omega$. We say that a play ω *satisfies* the objective Φ if $\omega \in \Phi$. We consider *ω-regular objectives* [25], specified as parity conditions. We also consider the special case of Büchi objectives.

- *Büchi objectives.* Let $B \subseteq V$ be a set of Büchi vertices. For a play $\omega = \langle v_0, v_1, \ldots \rangle \in \Omega$, we define $\mathrm{Inf}(\omega) = \{v \in V \mid v_k = v \text{ for infinitely many } k\}$ to be the set of vertices that occur infinitely often in ω. The Büchi objectives require that some vertex of B be visited infinitely often, and defines the set of winning plays $\text{Büchi}(B) = \{\omega \in \Omega \mid \mathrm{Inf}(\omega) \cap B \neq \emptyset\}$.
- *Parity objectives.* For $c, d \in \mathbb{N}$, we write $[c..d] = \{c, c+1, \ldots, d\}$. Let $p\colon V \to [0..d]$ be a function that assigns a *priority* $p(v)$ to every vertex $v \in V$, where $d \in \mathbb{N}$. The *parity objective* is defined as $\mathrm{Parity}(p) = \{\omega \in \Omega \mid \min\big(p(\mathrm{Inf}(\omega))\big) \text{ is even}\}$. In other words, the parity objective requires that the minimum priority visited infinitely often is even. In the sequel we will use Φ to denote parity objectives.

Qualitative analysis: almost-sure winning. Given a player-1 objective Φ, a strategy $\sigma \in \Sigma$ is *almost-sure winning* for player 1 from the vertex v if $\mathbb{P}_v^{\sigma}(\Phi) = 1$. The *almost-sure winning set* $\langle\langle 1 \rangle\rangle_{almost}(\Phi)$ for player 1 is the set of vertices from which player 1 has an almost-sure winning strategy. The qualitative analysis of MDPs corresponds to the computation of the almost-sure winning set for a given objective Φ.

Remark 1 *(Implication for parity objectives).* The almost-sure winning set for MDPs with parity objectives can be computed using $O(d)$ calls to compute the almost-sure winning set of MDPs with Büchi objectives [12,14,15,3,4,6]. Hence we focus on the qualitative analysis of MDPs with Büchi objectives. We will establish that the average case complexity is linear for Büchi objectives which implies an $O(m \cdot d)$ upper bound on the average case complexity for the qualitative analysis of MDPs with parity objectives, where m is the number of edges.

Algorithm for qualitative analysis. The algorithms for qualitative analysis for MDPs do not depend on the transition function, but only on the graph $G = ((V, E), (V_1, V_P))$. We now describe the classical algorithm for the qualitative analysis of MDPs with Büchi objectives. The algorithm requires the notion of random attractors.

Random attractor. Given an MDP G, let $U \subseteq V$ be a subset of vertices. The *random attractor* $Attr_P(U)$ is defined as follows: $X_0 = U$, and for $i \geq 0$, let $X_{i+1} = X_i \cup \{v \in V_P \mid E(v) \cap X_i \neq \emptyset\} \cup \{v \in V_1 \mid E(v) \subseteq X_i\}$. In other words, X_{i+1} consists of (a) vertices in X_i, (b) probabilistic vertices that have at least one edge to X_i, and (c) player-1 vertices, whose every successor is in X_i. Then $Attr_P(U) = \bigcup_{i \geq 0} X_i$. Observe that the random attractor is equivalent to the alternating reachability problem (reachability in AND-OR graphs).

Classical algorithm. The classical algorithm for MDPs with Büchi objectives is a simple iterative algorithm, and every iteration uses graph reachability and alternating graph reachability (random attractors). Let us denote the MDP in iteration i by G^i with vertex set V^i. Then in iteration i the algorithm executes the following steps: (i) computes the set Z^i of vertices that can reach the set of Büchi vertices $B \cap V^i$ in G^i; (ii) let $U^i = V^i \setminus Z^i$ be the set of remaining vertices; if U^i is empty, then the algorithm stops and outputs Z^i as the set of almost-sure winning vertices, and otherwise removes $Attr_P(U^i)$ from the graph, and continues to iteration $i+1$. The classical algorithm requires $O(n)$ iterations, where $n = |V|$, and each iteration requires $O(m)$ time, where $m = |E|$. Moreover the above analysis is tight, i.e., there exists a family of MDPs where the classical algorithm requires $\Omega(n)$ iterations, and total time $\Omega(n \cdot m)$. Hence $\Theta(n \cdot m)$ is the tight worst case complexity of the classical algorithm for MDPs with Büchi objectives. In this work we consider the average case analysis of the classical algorithm.

3. Average case analysis for MDPs with constant out-degree

In this section we consider the average case analysis of the number of iterations and the running time of the classical algorithm for computing the almost-sure winning set for MDPs with Büchi objectives on the families of graphs with constant out-degree (out-degree of every vertex fixed and bounded by two constants $d_{\min}$ and $d_{\max}$).

Family of graphs and results. We consider families of graphs where the vertex set V ($|V| = n$), the target set of Büchi vertices B ($|B| = t$), and the out-degree d_v of each vertex v is fixed across the whole family. The only varying component is the edges of the graph; for each vertex v, every set of vertices of size d_v is equally likely to be the neighbor set of v, independent of neighbors of other vertices. Finally, there exist constants $d_{\min}$ and $d_{\max}$ such that $d_{\min} \leq d_v \leq d_{\max}$ for all vertices v. We will show the following for this family of graphs: (a) if the target set B has size more than $30 \cdot x \cdot \log(n)$, where x is the number of distinct degrees, (i.e., $t \geq 30 \cdot x \cdot \log(n)$), then the expected number of iterations is $O(1)$ and the average running time is $O(n)$; and (b) if the target vertex set B has size at most $30 \cdot x \cdot \log(n)$, then the expected number of iterations required is $O(\log(n))$ and average running time is $O(n)$.

Notation. We use n and t for the total number of vertices and the size of the target set, respectively. We will denote by x the number of distinct out-degrees. Let d_i, for $1 \leq i \leq x$, be the distinct out-degrees. Since for all vertices v we have $d_{\min} \leq d_v \leq d_{\max}$, it follows that we have $x \leq d_{\max} - d_{\min} + 1$. Let a_i be the number of vertices with degree d_i and t_i be the number of target (Büchi) vertices with degree d_i.

The event $R(k_1,k_2,\dots,k_x)$. The *reverse reachable set* of the target set B is the set of vertices u such that there is a path in the graph from u to a vertex $v \in B$. Let S be any set comprising of k_i vertices of degree d_i, for $1 \le i \le x$. We define $R(k_1,k_2,\dots,k_x)$ as the probability of the event that all vertices of S can reach B via a path that lies entirely in S. Due to symmetry between vertices, this probability only depends on k_i, for $1 \le i \le x$ and is independent of S itself.[1] For ease of notation, we will sometimes denote the event itself by $R(k_1,k_2,\dots,k_x)$. We will investigate the reverse reachable set of B, which contains B itself. Recall that t_i vertices in B have degree d_i, and hence we are interested in the case when $k_i \ge t_i$ for all $1 \le i \le x$.

Consider a set S of vertices that is the reverse reachable set, and let S be composed of k_i vertices of degree d_i and of size k, i.e., $k = |S| = \sum_{i=1}^{x} k_i$. Since S is the reverse reachable set, it follows that for all vertices v in $V \setminus S$, there is no edge from v to a vertex in S (otherwise there would be a path from v to a target vertex and then v would belong to S). Thus there are no incoming edges from $V \setminus S$ to S. Thus for each vertex v of $V \setminus S$, all its neighbors must lie in $V \setminus S$ itself. This happens with probability $\prod_{i\in[1,x],a_i\neq k_i}\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i}$, since in $V \setminus S$ there are $a_i - k_i$ vertices with degree d_i and the size of $V \setminus S$ is $n-k$ (recall that $[1,x] = \{1,2,\dots,x\}$). Note that when $a_i \neq k_i$, there is at least one vertex of degree d_i in $V \setminus S$ that has all its neighbors in $V \setminus S$ and hence $n-k \ge d_i$. For simplicity of notation, we skip mentioning $a_i \neq k_i$ and substitute the term by 1 where $a_i = k_i$. The probability that each vertex in S can reach a target vertex is $R(k_1,k_2,\dots,k_x)$. Hence the probability of S being the reverse reachable set is given by:

$$\prod_{i=1}^{x}\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i} \cdot R(k_1,k_2,\dots,k_x)$$

There are $\prod_{i=1}^{x}\binom{a_i-t_i}{k_i-t_i}$ possible ways of choosing $k_i \ge t_i$ vertices (since the target set is contained) out of a_i. Notice that the terms are 1 where $a_i = k_i$. The value k can range from t to n and exactly one of these subsets of V will be the reverse reachable set. So the sum of probabilities of this happening is 1. Hence we have:

$$1 = \sum_{k=t}^{n}\ \sum_{\sum k_i=k,\, t_i\le k_i\le a_i}\left(\prod_{i=1}^{x}\binom{a_i-t_i}{k_i-t_i}\cdot\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i}\right)\cdot R(k_1,k_2,\dots,k_x) \tag{1}$$

Let

$$a_{k_1,k_2,\dots,k_x} = \left(\prod_{i=1}^{x}\binom{a_i-t_i}{k_i-t_i}\cdot\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i}\right)\cdot R(k_1,k_2,\dots,k_x);$$

$$\alpha_k = \sum_{\sum k_i=k,\, t_i\le k_i\le a_i} a_{k_1,k_2,\dots,k_x}.$$

Thus, $a_{k_1,k_2,\dots,k_x}$ is the probability that the reverse reachable set has exactly k_i vertices of degree d_i for $1 \le i \le x$, and α_k is the probability that the reverse reachable set has exactly k vertices.

Our goal is to show that for $30\cdot x\cdot\log(n) \le k \le n-1$, the value of α_k is very small; i.e., we want to get an upper bound on α_k. Note that two important terms in α_k are $\left(\binom{n-k}{d_i}/\binom{n}{d_i}\right)^{a_i-k_i}$ and $R(k_1,k_2,\dots,k_x)$. Below we get an upper bound for both of them. Firstly note that when k is small, for any set S comprising of k_i vertices of degree d_i for $1 \le i \le x$ and $|S| = k$, the event $R(k_1,k_2,\dots,k_x)$ requires each non-target vertex of S to have an edge inside S. Since k is small and all vertices have constant out-degree spread randomly over the entire graph, this is highly improbable. We formalize this intuitive argument in the following lemma.

Lemma 1 *(Upper bound on $R(k_1,k_2,\dots,k_x)$). For $k \le n - d_{\max}$*

$$R(k_1,k_2,\dots,k_x) \le \prod_{i=1}^{x}\left(1-\left(1-\frac{k}{n-d_i}\right)^{d_i}\right)^{k_i-t_i} \le \prod_{i=1}^{x}\left(\frac{d_i\cdot k}{n-d_{\max}}\right)^{k_i-t_i}.$$

Proof. Let S be the given set comprising of k_i vertices of degree d_i, for $1 \le i \le x$. Then for every non-target vertex of S, for it to be reachable to a target vertex via a path in S, it must have at least one edge inside S. This gives the following upper bound on $R(k_1,k_2,\dots,k_x)$.

[1] This holds because the out-degrees of vertices in S are fixed, but their neighbors are chosen randomly.

$$R(k_1, k_2, \ldots, k_x) \le \prod_{i=1}^{x} \left(1 - \frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{k_i - t_i}$$

We have the following inequality for all d_i, $1 \le i \le x$:

$$\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} = \prod_{j=0}^{d_i-1} \left(1 - \frac{k}{n-j}\right) \ge \left(1 - \frac{k}{n-d_i}\right)^{d_i} \ge 1 - \frac{d_i \cdot k}{n - d_i}$$

The first inequality follows by replacing j with $d_i \ge j$, and the second inequality follows from standard binomial expansion. Using the above inequality in the bound for $R(k_1, k_2, \ldots, k_x)$ we obtain

$$R(k_1, k_2, \ldots, k_x) \le \prod_{i=1}^{x} \left(1 - \left(1 - \frac{k}{n-d_i}\right)^{d_i}\right)^{k_i - t_i} \le \prod_{i=1}^{x} \left(\frac{d_i \cdot k}{n - d_i}\right)^{k_i - t_i} \le \prod_{i=1}^{x} \left(\frac{d_i \cdot k}{n - d_{\max}}\right)^{k_i - t_i}$$

The result follows. □

Now for $\left(\binom{n-k}{d_i}/\binom{n}{d_i}\right)^{a_i - k_i}$, we give an upper bound. First notice that when $a_i \ne k_i$, there is at least one vertex of degree d_i outside the reverse reachable set and it has all its edges outside the reverse reachable set. Hence, the size of the reverse reachable set (i.e. $n - k$) is at least d_i. Thus, $\binom{n-k}{d_i}$ is well defined.

Lemma 2. *For any $1 \le i \le x$ such that $a_i \ne k_i$, we have* $\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i - k_i} \le \left(1 - \frac{k}{n}\right)^{d_i \cdot (a_i - k_i)}$.

Proof. We have

$$\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i - k_i} = \left(\prod_{j=0}^{d_i-1} \left(1 - \frac{k}{n-j}\right)\right)^{a_i - k_i} \le \left(1 - \frac{k}{n}\right)^{d_i \cdot (a_i - k_i)}$$

The inequality follows since $j \ge 0$ and we replace j by 0 in the denominator. The result follows. □

Next we simplify the expression of α_k by taking care of the summation.

Lemma 3. *The probability that the reverse reachable set is of size exactly k is α_k, and*

$$\alpha_k \le n^x \cdot \max_{\sum k_i = k, t_i \le k_i \le a_i} a_{k_1, k_2, \ldots, k_x}.$$

Proof. The probability that the reverse reachable set is of size exactly k is given by

$$\alpha_k = \sum_{\sum k_i = k, t_i \le k_i \le a_i} \left(\prod_{i=1}^{x} \binom{a_i - t_i}{k_i - t_i} \cdot \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i - k_i}\right) \cdot R(k_1, k_2, \ldots, k_x)$$

(refer to Eq. (1)). Since

$$\alpha_k = \sum_{\sum k_i = k, t_i \le k_i \le a_i} a_{k_1, k_2, \ldots, k_x},$$

and there are x distinct degree's and n vertices, the number of different terms in the summation is at most n^x. Hence

$$\alpha_k \le n^x \cdot \max_{\sum k_i = k, t_i \le k_i \le a_i} a_{k_1, k_2, \ldots, k_x}.$$

The desired result follows. □

Now we proceed to achieve an upper bound on $a_{k_1, k_2, \ldots, k_x}$. First of all, intuitively if k is small, then $R(k_1, k_2, \ldots, k_x)$ is very small (this can be derived easily from Lemma 1). On the other hand, consider the case when k is very large. In this case there are very few vertices that cannot reach the target set. Hence they must have all their edges within them, which again has very low probability. Note that different factors that bind α_k depend on whether k is small or large. This suggests we should consider these cases separately. Our proof will consist of the following case analysis of the size k of the reverse reachable set: (1) Small k: $30 \cdot x \cdot \log(n) \le k \le c_1 \cdot n$ for some constant $c_1 > 0$, (2) Large k: $c_1 \cdot n \le k \le c_2 \cdot n$ for all constants $c_2 \ge c_1 > 0$, and (3) Very large k: $c_2 \cdot n \le k \le n - d_{\min} - 1$ for some constant $c_2 > 0$. The analysis of the constants will follow from the proofs. Note that since the target set B (with $|B| = t$) is a subset of its reverse reachable set, the case $k < t$ is infeasible. Hence in all the three cases, we will only consider $k \ge t$. We first consider the case when k is small.

3.1. Small k: $30 \cdot x \cdot \log(n) \le k \le c_1 n$

In this section we will consider the case when $30 \cdot x \cdot \log(n) \le k \le c_1 \cdot n$ for some constant $c_1 > 0$. Note that this case only occurs when $t \le c_1 \cdot n$ (since $k \ge t$). We will assume this throughout this section. We will prove that there exists a constant $c_1 > 0$ such that for all $30 \cdot x \cdot \log(n) \le k \le c_1 \cdot n$ the probability (α_k) that the size of the reverse reachable set is k is bounded by $\frac{1}{n^2}$. Note that we already have a bound on α_k in terms of $a_{k_1,k_2,\dots,k_x}$ (Lemma 3). We use continuous upper bounds of the discrete functions in $a_{k_1,k_2,\dots,k_x}$ to convert it into a form that is easy to analyze. Let

$$b_{k_1,k_2,\dots,k_x} = \prod_{i=1}^{x} \left(\frac{e \cdot (a_i - t_i)}{k_i - t_i} \right)^{k_i - t_i} \cdot e^{-\frac{k}{n} \cdot d_i \cdot (a_i - k_i)} \cdot \left(\frac{d_i \cdot k}{n - d_{\max}} \right)^{k_i - t_i},$$

where e is Euler's number (the base of the natural logarithm).

Lemma 4. *We have* $a_{k_1,k_2,\dots,k_x} \le b_{k_1,k_2,\dots,k_x}$.

Proof. We have

$$\begin{aligned} a_{k_1,k_2,\dots,k_x} &= \left(\prod_{i=1}^{x} \binom{a_i - t_i}{k_i - t_i} \cdot \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \right) \cdot R(k_1, k_2, \dots, k_x) \\ &\le \prod_{i=1}^{x} \binom{a_i - t_i}{k_i - t_i} \cdot \left(1 - \frac{k}{n} \right)^{d_i \cdot (a_i - k_i)} \cdot \left(\frac{d_i \cdot k}{n - d_{\max}} \right)^{k_i - t_i} \\ &\le \prod_{i=1}^{x} \left(\frac{e \cdot (a_i - t_i)}{k_i - t_i} \right)^{k_i - t_i} \cdot e^{-\frac{k}{n} d_i (a_i - k_i)} \cdot \left(\frac{d_i \cdot k}{n - d_{\max}} \right)^{k_i - t_i} \end{aligned}$$

The first inequality follows from Lemma 1 and Lemma 2. The second inequality follows from the first inequality of Proposition 1 (in Appendix A) and the fact that $1 - x \le e^{-x}$. □

Maximum of $b_{k_1,k_2,\dots,k_x}$. Next we show that $b_{k_1,k_2,\dots,k_x}$ drops exponentially as a function of k. Note that this is the reason for the logarithmic lower bound on k in this section. To achieve this we consider the maximum possible value achievable by $b_{k_1,k_2,\dots,k_x}$. Let $\partial_{k_i} b_{k_1,k_2,\dots,k_x}$ denote the change in $b_{k_1,k_2,\dots,k_x}$ due to change in k_i. For fixed $\sum_{i=1}^{x} k_i = k$, it is known that $b_{k_1,k_2,\dots,k_x}$ is maximized when for all i and j we have $\partial_{k_i} b_{k_1,k_2,\dots,k_x} = \partial_{k_j} b_{k_1,k_2,\dots,k_x}$. We have

$$\partial_{k_i} b_{k_1,k_2,\dots,k_x} = b_{k_1,k_2,\dots,k_x} \cdot \left(\frac{d_i \cdot k}{n} + \log\left(\frac{d_i \cdot k}{n - d_{\max}} \right) + \log\left(\frac{a_i - t_i}{k_i - t_i} \right) \right)$$

Thus, for maximizing $b_{k_1,k_2,\dots,k_x}$, for all i and j we must have

$$\begin{aligned} &\frac{d_i \cdot k}{n} + \log\left(\frac{d_i \cdot k}{n - d_{\max}} \right) + \log\left(\frac{a_i - t_i}{k_i - t_i} \right) = \frac{d_j \cdot k}{n} + \log\left(\frac{d_j \cdot k}{n - d_{\max}} \right) + \log\left(\frac{a_j - t_j}{k_j - t_j} \right) \\ &\Rightarrow \frac{k_i - t_i}{(a_i - t_i) \cdot \frac{d_i \cdot k}{n - d_{\max}} \cdot e^{d_i \cdot k/n}} = \frac{k_j - t_j}{(a_j - t_j) \cdot \frac{d_j \cdot k}{n - d_{\max}} \cdot e^{d_j \cdot k/n}} \\ &\Rightarrow \frac{k_i - t_i}{(a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}} = \frac{k_j - t_j}{(a_j - t_j) \cdot d_j \cdot e^{d_j \cdot k/n}} \end{aligned}$$

This implies that for all i we have

$$\begin{aligned} &\frac{k_i - t_i}{(a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}} = \frac{k - t}{\sum_{i=1}^{x} (a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}} \\ &\Rightarrow k_i - t_i = \frac{(a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}}{\sum_{i=1}^{x} (a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}} \cdot (k - t) \end{aligned}$$

Lemma 5. *Let* $L = \sum_{i=1}^{x}(a_i - t_i) \cdot d_i \cdot e^{d_i \cdot k/n}$. *We have*

$$b_{k_1,k_2,\dots,k_x} \le \left(\frac{L}{n-d_{\max}}\right)^{-t} \cdot \left(\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}\right)^{k}$$

Proof. The argument above shows that the maximum of $b_{k_1,k_2,\dots,k_x}$ is achieved when for all $1 \le i \le x$ we have $k_i - t_i = \frac{(a_i-t_i)\cdot d_i\cdot e^{d_i\cdot k/n}}{L} \cdot (k-t)$. Now, plugging the values in $b_{k_1,k_2,\dots,k_x}$, we get

$$b_{k_1,k_2,\dots,k_x} = \prod_{i=1}^{x}\left(\frac{e\cdot(a_i-t_i)}{k_i-t_i}\right)^{k_i-t_i} \cdot e^{-\frac{k}{n}\cdot d_i\cdot(a_i-k_i)} \cdot \left(\frac{d_i\cdot k}{n-d_{\max}}\right)^{k_i-t_i}$$

$$\le \prod_{i=1}^{x}\left(\frac{e\cdot L}{d_i\cdot e^{d_i\cdot k/n}\cdot(k-t)}\right)^{k_i-t_i} \cdot e^{-\frac{k}{n}\cdot d_i\cdot(a_i-k_i)} \cdot \left(\frac{d_i\cdot k}{n-d_{\max}}\right)^{k_i-t_i}$$

$$= \prod_{i=1}^{x}\left(\frac{L}{n-d_{\max}}\right)^{k_i-t_i} \cdot \left(e^{(k_i-t_i)} \cdot e^{-d_i\cdot(k/n)\cdot(k_i-t_i)} \cdot e^{-\frac{k}{n}\cdot d_i\cdot(a_i-k_i)}\right) \cdot \left(\frac{d_i\cdot k}{d_i\cdot(k-t)}\right)^{k_i-t_i}$$

(Rearranging denominators of first and third term, gathering powers of e together)

$$= \left(\frac{L}{n-d_{\max}}\right)^{\sum_{i=1}^{x}(k_i-t_i)} \cdot \left(e^{\sum_{i=1}^{x}(k_i-t_i)} \cdot e^{-\sum_{i=1}^{x} d_i\cdot(k/n)\cdot(a_i-t_i)}\right) \cdot \left(\frac{k}{(k-t)}\right)^{\sum_{i=1}^{x}(k_i-t_i)}$$

(Product is transformed to sum in exponent)

$$= \left(\frac{L}{n-d_{\max}}\right)^{(k-t)} \cdot \left(e^{(k-t)} \cdot e^{-(k/n)\cdot\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}\right) \cdot \left(1+\frac{t}{(k-t)}\right)^{(k-t)}$$

(As $\sum_{i=1}^{x} k_i - t_i = k - t$)

$$\le \left(\frac{L}{n-d_{\max}}\right)^{k-t} \cdot e^{k-t} \cdot e^{-k/n\cdot\sum_{i=1}^{x} d_i\cdot(a_i-t_i)} \cdot e^{t}$$

(Since $1+x \le e^x$ we have $\left(1+\frac{t}{k-t}\right) \le e^{\frac{t}{k-t}}$)

$$= \left(\frac{L}{n-d_{\max}}\right)^{-t} \cdot \left(\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}\right)^{k}$$

(Arranging in powers by t and k).

The desired result follows. □

We now establish an upper bound on each term in the bound of Lemma 5. First, we consider the term $\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}$.

Lemma 6. *Let n be sufficiently large and let* $c_1 \le \frac{0.04}{d_{\max}}$. *Then for all* $k \le c_1 \cdot n$ *we have* $\left(\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}\right) \le \frac{9}{10}$.

Proof. We have the following inequality:

$$\left(\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}\right) = \frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)\cdot e^{d_i\cdot k/n}}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}$$

$$\le \frac{e^{d_{\max}\cdot c_1}}{n-d_{\max}} \cdot \left(\sum_{i=1}^{x} d_i\cdot(a_i-t_i)\right) \cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}$$

($d_i \le d_{\max}$ and $k \le c_1 \cdot n$)

$$\le e^{d_{\max}\cdot c_1}\cdot\frac{n}{n-d_{\max}}\cdot\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}\cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}$$

(multiplying numerator and denominator with n)

$$= e^{d_{\max}\cdot c_1}\cdot\frac{n}{n-d_{\max}}\cdot\frac{d}{e^{d-1}}$$

Here,

$$d=\frac{1}{n}\cdot\sum_{i=1}^{x} d_i\cdot(a_i-t_i)\ge d_{\min}\cdot\frac{n-t}{n}\ge d_{\min}\cdot(1-c_1)\ge 1$$

The last inequality follows because $c_1\le 0.5$ and $d_{\min}\ge 2$. Since $f(d)=d/e^{d-1}$ is a decreasing function for $d\ge 1$, we have $f(d)\le f(d_{\min}\cdot(1-c_1))$. Thus,

$$\begin{aligned}
\frac{e^{d_{\max}\cdot c_1}\cdot n}{n-d_{\max}}\cdot\frac{d}{e^{d-1}} &\le e^{d_{\max}\cdot c_1}\cdot\frac{n}{n-d_{\max}}\cdot\frac{d_{\min}\cdot(1-c_1)}{e^{d_{\min}\cdot(1-c_1)-1}}\\
&= e^{(d_{\min}+d_{\max})\cdot c_1}\cdot\frac{n}{n-d_{\max}}\cdot\frac{d_{\min}\cdot(1-c_1)}{e^{d_{\min}-1}}\\
&\le e^{2\cdot d_{\max}\cdot c_1}\cdot\frac{n}{n-d_{\max}}\cdot\frac{2}{e}\quad (1-c_1\le 1 \text{ and } f(d_{\min})\le f(2)=2/e)\\
&\le 2\cdot e^{-0.92}\cdot\frac{1}{0.9}\quad\left(\frac{n}{n-d_{\max}}\le\frac{1}{0.9}\text{ for sufficiently large } n \text{ and } c_1\le\frac{0.04}{d_{\max}}\right)\\
&\le 0.9
\end{aligned}$$

The desired result follows. □

Finally, we provide an upper bound on the remaining term $\frac{L}{n-d_{\max}}$ in the bound of Lemma 5.

Lemma 7. *For sufficiently large n and $c_1\le 0.2$ we have $\frac{L}{n-d_{\max}}\ge 1$.*

Proof. We have the following inequality:

$$\begin{aligned}
L &= \sum_{i=1}^{x}(a_i-t_i)\cdot d_i\cdot e^{d_i\cdot k/n}\\
&\ge 2\cdot\sum_{i=1}^{x}(a_i-t_i)\\
&= 2\cdot(n-t)\\
&\ge 2\cdot n\cdot(1-c_1)\\
&\ge 1.6\cdot n,
\end{aligned}$$

where the second transition holds because $e^{d_i\cdot k/n}\ge 1$ and $d_i\ge d_{\min}\ge 2$, the fourth transition holds because $t\le c_1\cdot n$, and the last transition holds because $c_1\le 0.2$. Finally, $n-d_{\max}<1.6\cdot n$ for large n. Hence, the desired result follows. □

Now we prove a bound on $b_{k_1,k_2,\ldots,k_x}$.

Lemma 8 *(Upper bound on $b_{k_1,k_2,\ldots,k_x}$). There exists a constant $c_1>0$ such that for sufficiently large n and $t\le k\le c_1\cdot n$, we have $b_{k_1,k_2,\ldots,k_x}\le\left(\frac{9}{10}\right)^k$.*

Proof. Let $0<c_1\le\frac{0.04}{d_{\max}}\le 0.2$ as in Lemma 6. By Lemma 5 we have

$$b_{k_1,k_2,\ldots,k_x}\le\left(\frac{L}{n-d_{\max}}\right)^{-t}\cdot\left(\frac{L}{n-d_{\max}}\cdot e^{1-\frac{\sum_{i=1}^{x} d_i\cdot(a_i-t_i)}{n}}\right)^k$$

By Lemma 7 we have $\left(\frac{L}{n-d_{\max}}\right)\ge 1$, and hence $\left(\frac{L}{n-d_{\max}}\right)^{-t}\le 1$. By Lemma 6 we have

$$\frac{L}{n-d_{\max}} \cdot e^{1-\frac{\sum_{i=1}^{x} d_i \cdot (a_i - t_i)}{n}} \leq \frac{9}{10}$$

The desired result follows trivially. □

Taking appropriate bounds on the value of k, we get an upper bound on $a_{k_1,k_2,\ldots,k_x}$. Recall that x is the number of distinct degrees and hence $x \leq d_{\max} - d_{\min} + 1$.

Lemma 9 *(Upper bound on $a_{k_1,k_2,\ldots,k_x}$). There exists a constant $c_1 > 0$ such that for sufficiently large n with $t \leq c_1 \cdot n$ and for all $30 \cdot x \cdot \log(n) \leq k \leq c_1 \cdot n$, we have $a_{k_1,k_2,\ldots,k_x} < \frac{1}{n^{3 \cdot x}}$.*

Proof. By Lemma 4 we have $a_{k_1,k_2,\ldots,k_x} \leq b_{k_1,k_2,\ldots,k_x}$ and by Lemma 8 we have $b_{k_1,k_2,\ldots,k_x} \leq \left(\frac{9}{10}\right)^k$. Thus for $k \geq 30 \cdot x \cdot \log(n)$,

$$a_{k_1,k_2,\ldots,k_x} \leq \left(\frac{9}{10}\right)^{30 \cdot x \cdot \log(n)} = n^{30 \cdot x \cdot \log(9/10)} \leq \frac{1}{n^{3 \cdot x}}$$

The desired result follows. □

Lemma 10 *(Main lemma for small k). There exists a constant $c_1 > 0$ such that for sufficiently large n with $t \leq c_1 \cdot n$ and for all $30 \cdot x \cdot \log(n) \leq k \leq c_1 \cdot n$, the probability that the size of the reverse reachable set S is k is at most $\frac{1}{n^2}$.*

Proof. The probability that the reverse reachable set is of size k is given by α_k. By Lemma 3 and Lemma 9 it follows that the probability is at most $n^x \cdot n^{-3 \cdot x} = n^{-2 \cdot x} \leq \frac{1}{n^2}$. The desired result follows. □

3.2. Large k: $c_1 \cdot n \leq k \leq c_2 \cdot n$

In this section we will show that for all constants c_1 and c_2, with $0 < c_1 \leq c_2$, when $t \leq c_2 \cdot n$ the probability α_k is at most $\frac{1}{n^2}$ for all $c_1 \cdot n \leq k \leq c_2 \cdot n$. We start with some notation that we will use in the proofs. Let $a_i = p_i \cdot n$, $t_i = y_i \cdot n$, $k_i = s_i \cdot n$ for $1 \leq i \leq x$ and $k = s \cdot n$ for $c_1 \leq s < c_2$. We first present a bound on $a_{k_1,k_2,\ldots,k_x}$.

Lemma 11. *For all constants c_1 and c_2 with $0 < c_1 \leq c_2$ and for all $c_1 \cdot n \leq k \leq c_2 \cdot n$, we have*

$$a_{k_1,k_2,\ldots,k_x} \leq (n+1)^x \cdot \mathrm{Term}_1 \cdot \mathrm{Term}_2,$$

where

$$\mathrm{Term}_1 = \left(\prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i}\right)^{s_i - y_i} \left(\frac{p_i - y_i}{p_i - s_i}\right)^{p_i - s_i} (1-s)^{d_i(p_i - s_i)} (1 - (1-s)^{d_i})^{s_i - y_i} \right)^n$$

and

$$\mathrm{Term}_2 = \prod_{i=1}^{x} \left(\frac{1 - \left(1 - \frac{s}{1 - d_i/n}\right)^{d_i}}{1 - (1-s)^{d_i}} \right)^{n(s_i - y_i)}.$$

Proof. We have

$$\begin{aligned}
a_{k_1,k_2,\ldots,k_x} &= \left(\prod_{i=1}^{x} \binom{a_i - t_i}{k_i - t_i} \cdot \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \right) \cdot R(k_1, k_2, \ldots, k_x) \\
&\leq \left(\prod_{i=1}^{x} (a_i - t_i + 1) \cdot \left(\frac{a_i - t_i}{k_i - t_i}\right)^{k_i - t_i} \cdot \left(\frac{a_i - t_i}{a_i - k_i}\right)^{a_i - k_i} \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \right) \cdot R(k_1, k_2, \ldots, k_x) \\
&\qquad \text{(Applying second inequality of Proposition 1 with } \ell = a_i - t_i \text{ and } j = k_i - t_i) \\
&\leq (n+1)^x \cdot \left(\prod_{i=1}^{x} \left(\frac{a_i - t_i}{k_i - t_i}\right)^{k_i - t_i} \cdot \left(\frac{a_i - t_i}{a_i - k_i}\right)^{a_i - k_i} \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \right) \cdot R(k_1, k_2, \ldots, k_x).
\end{aligned}$$

Proposition 1 is presented in Appendix A. The last inequality above is obtained as follows: $(a_i - t_i + 1) \leq n + 1$ as $a_i \leq n$. Our goal is now to show that

$$Y = \left(\prod_{i=1}^{x} \left(\frac{a_i - t_i}{k_i - t_i} \right)^{k_i - t_i} \cdot \left(\frac{a_i - t_i}{a_i - k_i} \right)^{a_i - k_i} \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \right) \cdot R(k_1, k_2, \ldots, k_x) \leq \mathsf{Term}_1 \cdot \mathsf{Term}_2.$$

We have (i) $a_i - t_i = n(p_i - y_i)$; (ii) $k_i - t_i = n(s_i - y_i)$; and (iii) $a_i - k_i = n(p_i - s_i)$. Hence we have

$$\prod_{i=1}^{x} \left(\frac{a_i - t_i}{k_i - t_i} \right)^{k_i - t_i} \cdot \left(\frac{a_i - t_i}{a_i - k_i} \right)^{a_i - k_i} = \prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i} \right)^{n(s_i - y_i)} \left(\frac{p_i - y_i}{p_i - s_i} \right)^{n(p_i - s_i)}.$$

By Lemma 2 we have

$$\prod_{i=1}^{x} \left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}} \right)^{a_i - k_i} \leq \prod_{i=1}^{x} \left(1 - \frac{k}{n} \right)^{d_i \cdot n \cdot (p_i - s_i)}$$

By Lemma 1 we have

$$R(k_1, k_2, \ldots, k_x) \leq \prod_{i=1}^{x} \left(1 - \left(1 - \frac{k}{n - d_i} \right)^{d_i} \right)^{n(s_i - y_i)}$$

Hence we have

$$\begin{aligned}
Y &\leq \prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i} \right)^{n(s_i - y_i)} \left(\frac{p_i - y_i}{p_i - s_i} \right)^{n(p_i - s_i)} \left(1 - \frac{k}{n} \right)^{d_i n (p_i - s_i)} \left(1 - \left(1 - \frac{k}{n - d_i} \right)^{d_i} \right)^{n(s_i - y_i)} \\
&= \prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i} \right)^{n(s_i - y_i)} \left(\frac{p_i - y_i}{p_i - s_i} \right)^{n(p_i - s_i)} (1 - s)^{d_i n (p_i - s_i)} \left(1 - \left(1 - \frac{s}{1 - d_i/n} \right)^{d_i} \right)^{n(s_i - y_i)} \\
&= \underbrace{\left(\prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i} \right)^{s_i - y_i} \left(\frac{p_i - y_i}{p_i - s_i} \right)^{p_i - s_i} (1 - s)^{d_i (p_i - s_i)} \right)^{n}}_{X_1} \cdot \underbrace{\prod_{i=1}^{x} \left(1 - \left(1 - \frac{s}{1 - d_i/n} \right)^{d_i} \right)^{n(s_i - y_i)}}_{X_2} \\
&= \left(\prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i} \right)^{s_i - y_i} \left(\frac{p_i - y_i}{p_i - s_i} \right)^{p_i - s_i} (1 - s)^{d_i (p_i - s_i)} (1 - (1 - s)^{d_i})^{s_i - y_i} \right)^{n} \\
&\quad \cdot \prod_{i=1}^{x} \left(\frac{1 - \left(1 - \frac{s}{1 - d_i/n} \right)^{d_i}}{1 - (1 - s)^{d_i}} \right)^{n(s_i - y_i)}
\end{aligned}$$

The last equality is obtained by multiplying $(1 - (1 - s)^{d_i})^{n(s_i - y_i)}$ to X_1 and dividing it from X_2. Thus we obtain $Y \leq \mathsf{Term}_1 \cdot \mathsf{Term}_2$, and the result follows. $\square$

Given the bound in Lemma 11, we now present upper bounds on Term_2 and Term_1.

Lemma 12. *Term_2 of Lemma 11, i.e., $\prod_{i=1}^{x} \left(\frac{1 - \left(1 - \frac{s}{1 - d_i/n} \right)^{d_i}}{1 - (1 - s)^{d_i}} \right)^{n(s_i - y_i)}$ is bounded from above by a constant.*

Proof. We have

$$\begin{aligned}
\left(\frac{1 - \left(1 - \frac{s}{1 - d_i/n} \right)^{d_i}}{1 - (1 - s)^{d_i}} \right)^{n(s_i - y_i)} &\leq \left(\frac{1 - \left(1 - s(1 + \frac{2d_i}{n}) \right)^{d_i}}{1 - (1 - s)^{d_i}} \right)^{n(s_i - y_i)} \quad \text{(for sufficiently large } n\text{)} \\
&\leq \left(\frac{1 - (1 - s)^{d_i} + \binom{d_i}{1} \cdot \frac{2sd_i}{n} \cdot (1 - s)^{d_i - 1}}{1 - (1 - s)^{d_i}} \right)^{n(s_i - y_i)}
\end{aligned}$$

(taking first two terms of binomial expansion)

$$= \left(1 + \frac{\frac{(1-s)^{d_i-1}}{1-(1-s)^{d_i}} \cdot 2sd_i{}^2}{n}\right)^{n(s_i-y_i)}$$

$$\le e^{\frac{(1-s)^{d_i-1}}{1-(1-s)^{d_i}} \cdot 2sd_i{}^2 \cdot (s_i-y_i)} \quad ((1+x) \le e^x).$$

Since $c_1 \le s \le c_2$ we have s is constant, and similarly $d_{\min} \le d_i \le d_{\max}$ and hence d_i is constant. Hence it follows that the above expression is constant and hence the product of those terms for $1 \le i \le x$ is also bounded by a constant (since x is constant). The result follows. □

Lemma 13. *There exists a constant $0 < \eta < 1$ such that* Term_1 *of Lemma 11 is at most η^n (exponentially small), i.e.,*

$$\left(\prod_{i=1}^{x} \left(\frac{p_i - y_i}{s_i - y_i}\right)^{s_i - y_i} \left(\frac{p_i - y_i}{p_i - s_i}\right)^{p_i - s_i} (1-s)^{d_i(p_i - s_i)} (1-(1-s)^{d_i})^{s_i - y_i}\right)^n \le \eta^n$$

Proof. Let

$$f(d_i) = \left(\frac{p_i - y_i}{s_i - y_i}\right)^{s_i - y_i} \left(\frac{p_i - y_i}{p_i - s_i}\right)^{p_i - s_i} (1-s)^{d_i(p_i - s_i)} (1-(1-s)^{d_i})^{s_i - y_i}$$

Note that $f(d_i)$ is maximum when

$$\partial_{d_i} f(d_i) = 0 \Leftrightarrow d_i^* = \frac{\log\left(\frac{p_i - s_i}{p_i - y_i}\right)}{\log(1-s)}$$

Moreover, it can easily be checked that this maximum value is $f(d_i^*) = 1$. Hence, in general we have $f(d_i) \le 1$. We wish to prove that there exists some i such that $d_i \ne d_i^*$. Suppose for contradicton that $d_i = d_i^*$ for all i. Then, we have

$$d_i^* \ge 2 \Rightarrow (1-s)^2 \ge \frac{p_i - s_i}{p_i - y_i}$$

for all i. For fractions α_i/β_i, we have $(\sum_i \alpha_i)/(\sum_i \beta_i) \le \max_i \alpha_i/\beta_i$. Hence, we have

$$(1-s)^2 \ge \frac{\sum_i (p_i - s_i)}{\sum_i (p_i - y_i)} = \frac{1-s}{1-y} \Rightarrow (1-s)(1-y) \ge 1$$

The last inequality is a contradiction, because $0 < s < 1$. Hence, not all d_i can be equal to d_i^*. Hence, $\prod_i f(d_i)$ cannot achieve its maximum value 1. Since each $d_{i^*} \in [d_{\min}, d_{\max}]$ has a compact domain and f is a continuous function, there exists a constant $\eta < 1$ such that $\prod_{i=1}^{x} f(d_i) \le \eta$. The result thus follows. □

Lemma 14 *(Main lemma for large k). For all constants c_1 and c_2 with $0 < c_1 \le c_2$, when n is sufficiently large and $t \le c_2 \cdot n$, for all $c_1 \cdot n \le k \le c_2 \cdot n$, the probability that the size of the reverse reachable set S is k is at most $\frac{1}{n^2}$.*

Proof. By Lemma 11, we have $a_{k_1,k_2,\dots,k_x} \le (n+1)^x \cdot \text{Term}_1 \cdot \text{Term}_2$, and by Lemma 12 and Lemma 13, Term_2 is a constant and Term_1 is exponentially small in n, where $x \le (d_{\max} - d_{\min} + 1)$. The exponentially small Term_1 overrides the polynomial factor $(n+1)^x$ and the constant Term_2, and ensures that $a_{k_1,k_2,\dots,k_x} \le n^{-3x}$. By Lemma 3 it follows that $\alpha_k \le n^{-2x} \le \frac{1}{n^2}$. □

3.3. *Very large k: $(1 - 1/e^2)n$ to $n - d_{\min} - 1$*

In this subsection we consider the case when the size k of the reverse reachable set is between $(1 - \frac{1}{e^2}) \cdot n$ and $n - d_{\min} - 1$. Note that if the reverse reachable set has size at least $n - d_{\min}$, then the reverse reachable set must be the set of all vertices, as otherwise the remaining vertices cannot have enough edges among themselves. Take $\ell = n - k$. Hence $d_{\min} + 1 \le \ell \le n/e^2$. As stated earlier, in this case $a_{k_1,k_2,\dots,k_x}$ becomes small since we require that the ℓ vertices outside the reverse reachable set must have all their edges within themselves; this corresponds to the factor of $\left(\binom{n-k}{d_i}/\binom{n}{d_i}\right)^{a_i - k_i}$. Since ℓ is very small, this has a very low probability. With this intuition, we proceed to show the following bound on $a_{k_1,k_2,\dots,k_x}$.

Lemma 15. *We have $a_{k_1,k_2,\dots,k_x} \le \left(x \cdot e \cdot \frac{\ell}{n}\right)^{\ell}$.*

Proof. We have

$$\begin{aligned}
a_{k_1,k_2,\dots,k_x} &= \left(\prod_{i=1}^{x}\binom{a_i-t_i}{k_i-t_i}\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i}\right)\cdot R(k_1,k_2,\dots,k_x)\\
&\le \prod_{i=1}^{x}\binom{a_i-t_i}{k_i-t_i}\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i} \qquad \text{(Ignoring probability value } R(k_1,k_2,\dots,k_x)\le 1)\\
&= \prod_{i=1}^{x}\binom{a_i-t_i}{a_i-k_i}\left(\frac{\binom{n-k}{d_i}}{\binom{n}{d_i}}\right)^{a_i-k_i} \qquad \left(\text{Since } \binom{x}{y}=\binom{x}{x-y}\right)\\
&\le \prod_{i=1}^{x}\binom{a_i-t_i}{a_i-k_i}\left(1-\frac{k}{n}\right)^{d_i(a_i-k_i)} \qquad \text{(By Lemma 2)}\\
&\le \prod_{i=1}^{x}\left(\frac{e\cdot(a_i-t_i)}{a_i-k_i}\right)^{a_i-k_i}\left(\frac{n-k}{n}\right)^{d_i(a_i-k_i)} \qquad \text{(Inequality 1 of Proposition 1)}\\
&\le e^{\ell}\cdot\left(\frac{\ell}{n}\right)^{2\ell}\cdot\prod_{i=1}^{x}\left(\frac{a_i-t_i}{a_i-k_i}\right)^{a_i-k_i} \qquad \left(\text{Since } d_i\ge 2 \text{ and } \sum_{i=1}^{x}(a_i-k_i)=\ell\right)
\end{aligned}$$

Recall that in the product appearing in the last expression, we take the value of the term to be 1 where $a_i = k_i$. Proposition 1 is presented in Appendix A. Since for all i we have $(a_i - t_i) \le n - t$, it follows that $\prod_{i=1}^{x}(a_i-t_i)^{a_i-k_i} \le \prod_{i=1}^{x}(n-t)^{a_i-k_i} = (n-t)^{\ell}$.

We also want a lower bound for $\prod_{i=1}^{x}(a_i-k_i)^{a_i-k_i}$. Note that $\sum_{i=1}^{x}(a_i-k_i)=\ell$ is fixed. Hence, this is a problem of minimizing $\prod_{i=1}^{x} {y_i}^{y_i}$ given that $\sum_{i=1}^{x} y_i = \ell$ is fixed. As before, this reduces to $\partial_{y_a}\prod_{i=1}^{x} {y_i}^{y_i} = \partial_{y_b}\prod_{i=1}^{x} {y_i}^{y_i}$, for all a, b. Hence, the minimum is attained at $y_i = \ell/x$, for all i. Hence, $\prod_{i=1}^{x}(a_i-k_i)^{a_i-k_i} \ge \left(\frac{\ell}{x}\right)^{\ell}$. Combining these,

$$\begin{aligned}
a_{k_1,k_2,\dots,k_x} &\le e^{\ell}\cdot\left(\frac{\ell}{n}\right)^{2\ell}\cdot\prod_{i=1}^{x}\left(\frac{a_i-t_i}{a_i-k_i}\right)^{a_i-k_i}\\
&\le e^{\ell}\cdot\left(\frac{\ell}{n}\right)^{2\ell}\cdot\left(\frac{n-t}{\left(\frac{\ell}{x}\right)}\right)^{\ell}\\
&\le \left(x\cdot e\cdot\frac{\ell}{n}\right)^{\ell}
\end{aligned}$$

Hence we have the desired inequality. □

We see that $\left(x\cdot e\cdot\frac{\ell}{n}\right)^{\ell}$ is a convex function in ℓ and its maximum is attained at one of the endpoints. For $\ell = n/e^2$, the bound is exponentially decreasing with n whereas for constant ℓ, the bound is polynomially decreasing in n. Hence, the maximum is attained at left endpoint of the interval (constant value of ℓ). However, the bound we get is not sufficient to apply Lemma 3 directly. We break this case into two sub-cases; $d_{\max}+1 < \ell \le n/e^2$ and $d_{\min}+1 \le \ell \le d_{\max}+1$.

Lemma 16. *For $d_{\max}+1 < \ell \le n/e^2$, we have $a_{k_1,k_2,\dots,k_x} < n^{-(2+x)}$ and $\alpha_k \le 1/n^2$.*

Proof. As we have seen, we only need to prove this for the value of ℓ where $a_{k_1,k_2,\dots,k_x}$ attains its maximum i.e. $\ell = d_{\max}+2$. Note that $d_{\max}+1 = x + d_{\min} \ge x+2$. Hence,

$$\begin{aligned}
a_{k_1,k_2,\dots,k_x} &\le \left(x\cdot e\cdot\frac{\ell}{n}\right)^{\ell} \qquad \text{(By Lemma 15)}\\
&\le \left(x\cdot e\cdot\frac{d_{\max}+2}{n}\right)^{d_{\max}+2}\\
&= (x\cdot e\cdot(d_{\max}+2))^{d_{\max}+2}\cdot n^{-(d_{\max}+2)}\\
&< n^{-(d_{\max}+1)} \qquad \text{(Since first term is a constant)}\\
&\le n^{-(2+x)}
\end{aligned}$$

Hence we obtain the first inequality of the lemma. By Lemma 3 and the first inequality of the lemma we have $\alpha_k \le \frac{1}{n^2}$. □

Lemma 17. *There exists a constant $h > 0$ such that for $d_{\min} + 1 \le \ell \le d_{\max} + 1$, we have $a_{k_1,k_2,\ldots,k_x} < h \cdot n^{-\ell}$ and $\alpha_k \le \frac{h}{n^2}$.*

Proof. By Lemma 15 we have

$$a_{k_1,k_2,\ldots,k_x} \le \left(x \cdot e \cdot \frac{\ell}{n}\right)^{\ell}$$
$$\le (x \cdot e \cdot (d_{\max} + 1))^{d_{\max}+1} \cdot n^{-\ell}$$

Let $h = (x \cdot e \cdot (d_{\max} + 1))^{d_{\max}+1}$. Hence, first part is proved.

Now, for the second part, we note that since there are ℓ vertices outside the reverse reachable set, and all their edges must be within these ℓ vertices, they must have degree at most $\ell - 1$. Hence, there are now n vertices with at most $\ell - d_{\min}$ distinct degrees. Hence, in the summation

$$\alpha_k = \sum_{\substack{k_1,\ldots,k_x \text{ s.t.} \\ \sum k_i = k, t_i \le k_i \le a_i}} a_{k_1,k_2,\ldots,k_x},$$

there are at most $n^{\ell - d_{\min}}$ terms. Thus we have

$$\alpha_k \le n^{\ell - d_{\min}} \cdot h \cdot n^{-\ell} = h \cdot n^{-d_{\min}} \le \frac{h}{n^2}.$$

The desired result follows. □

Lemma 18 *(Main lemma for very large k). For all t, for all $(1 - \frac{1}{e^2}) \cdot n \le k \le n - 1$, the probability that the size of the reverse reachable set S is k is at most $O(\frac{1}{n^2})$.*

Proof. By Lemma 16 and Lemma 17 we obtain the result for all $(1 - \frac{1}{e^2}) \cdot n \le k \le n - d_{\min} - 1$. Since the reverse reachable set must contain all vertices if it has size at least $n - d_{\min}$, the result follows. □

3.4. Expected number of iterations and running time

From Lemma 10, Lemma 14, and Lemma 18, we obtain that there exists a constant h such that

$$\alpha_k \le \frac{1}{n^2}, \qquad 30 \cdot x \cdot \log(n) \le k < n - d_{\max} - 1$$
$$\alpha_k \le \frac{h}{n^2}, \qquad n - d_{\max} - 1 \le k \le n - d_{\min} - 1$$
$$\alpha_k = 0 \qquad n - d_{\min} \le k \le n - 1$$

Hence using the union bound we get the following result

Lemma 19 *(Lemma for size of the reverse reachable set). $\mathbb{P}(|S| < 30 \cdot x \cdot \log(n)$ or $|S| = n) \ge 1 - \frac{h}{n}$, where S is the reverse reachable set of target set (i.e., with probability at least $1 - \frac{h}{n}$ either at most $30 \cdot x \cdot \log(n)$ vertices reach the target set or all the vertices reach the target set).*

Proof.

$$\begin{aligned}
\mathbb{P}(|S| < 30 \cdot x \cdot \log(n) \text{ or } |S| = n) &= 1 - \mathbb{P}(30 \cdot x \cdot \log(n) \le |S| \le n - 1) \\
&\ge 1 - \frac{n - d_{\max} - 1}{n^2} - \frac{h(d_{\max} - d_{\max})}{n^2} - 0 \\
&\ge 1 - \frac{h(n - d_{\max} - 1)}{n^2} - \frac{h(d_{\max} - d_{\max})}{n^2} \\
&\ge 1 - \frac{hn}{n^2} \\
&= 1 - \frac{h}{n} \qquad \square
\end{aligned}$$

In addition, we note that the number of iterations of the classical algorithm is bounded by the size of the reverse reachable set, because after the first iteration, the graph is reduced to the sub-graph induced by the reverse reachable set.

Let $I(n)$ and $T(n)$ denote the expected number of iterations and the expected running time of the classical algorithm for MDPs on random graphs with n vertices and constant out-degree. Then from above we have

$$I(n) \leq \left(1 - \frac{h}{n}\right) \cdot 30 \cdot x \cdot \log(n) + \frac{h}{n} \cdot n$$

It follows that $I(n) = O(\log(n))$. For the expected running time we have

$$T(n) \leq \left(1 - \frac{h}{n}\right) \cdot (30 \cdot x \cdot \log(n))^2 + \frac{h}{n} \cdot n^2$$

It follows that $T(n) = O(n)$. Hence we have the following theorem.

Theorem 1. *The expected number of iterations and the expected running time of the classical algorithm for MDPs with Büchi objectives over graphs with constant out-degree are $O(\log(n))$ and $O(n)$, respectively.*

Remark 2. For Theorem 1, we considered the model where the out-degree of each vertex v is fixed as d_v and there exist constants $d_{\min}$ and $d_{\max}$ such that $d_{\min} \leq d_v \leq d_{\max}$ for every vertex v. We discuss the implication of Theorem 1 for related models. First, when the out-degrees of all vertices are same and constant (say d^*), Theorem 1 can be applied with the special case of $d_{\min} = d_{\max} = d^*$. A second possible alternative model is when the out-degree of every vertex is a distribution over the range $[d_{\min}, d_{\max}]$. Since we proved that the average case is linear for every possible value of the out-degree d_v in $[d_{\min}, d_{\max}]$ for every vertex v (i.e., for all possible combinations), it implies that the average case is also linear when the out-degree is a distribution over $[d_{\min}, d_{\max}]$.

4. Average case analysis in Erdös–Rényi model

In this section we consider the classical Erdös–Rényi model of random graphs $\mathcal{G}_{n,p}$, with n vertices, where each edge is chosen to be in the graph independently with probability p [17] (we consider directed graphs and then $\mathcal{G}_{n,p}$ is also referred as $\mathcal{D}_{n,p}$ in the literature). First, in Section 4.1 we consider the case when p is $\Omega\left(\frac{\log(n)}{n}\right)$, and then we consider the case when $p = \frac{1}{2}$ (that generates the uniform distribution over all graphs). We will show two results: (1) if $p \geq \frac{c \cdot \log(n)}{n}$, for some constant $c > 2$, then the expected number of iterations is constant and the expected running time is linear; and (2) if $p = \frac{1}{2}$ (with $p = \frac{1}{2}$ we consider all graphs to be equally likely), then the probability that the number of iterations is more than one falls exponentially in n (in other words, graphs where the running time is more than linear are exponentially rare).

4.1. $\mathcal{G}_{n,p}$ with $p = \Omega\left(\frac{\log(n)}{n}\right)$

In this subsection we will show that given $p \geq \frac{c \cdot \log(n)}{n}$, for some constant $c > 2$, the probability that not all vertices can reach the given target set is $O(1/n)$. Hence the expected number of iterations of the classical algorithm for MDPs with Büchi objectives is constant and hence the algorithm works in average time linear in the size of the graph. Observe that to show the result the worst possible case is when the size of the target set is 1, as otherwise the chance that all vertices reach the target set is higher. Thus from here onwards, we assume that the target set has exactly 1 vertex.

The probability $R(n, p)$. For a random graph in $\mathcal{G}_{n,p}$ and a given target vertex, we denote by $R(n, p)$ the probability that each vertex in the graph has a path along the directed edges to the target vertex. Our goal is to obtain a lower bound on $R(n, p)$.

The key recurrence. Consider a random graph G with n vertices, with a given target vertex, and edge probability p. For a set K of vertices with size k (i.e., $|K| = k$), which contains the target vertex, $R(k, p)$ is the probability that each vertex in the set K, has a path to the target vertex, that lies within the set K (i.e., the path only visits vertices in K). The probability $R(k, p)$ depends only on k and p, due to the symmetry among vertices.

Consider the subset S of all vertices in V, which have a path to the target vertex. In that case, for all vertices v in $V \setminus S$, there is no edge going from v to a vertex in S (otherwise there would have been a path from v to the target vertex). Thus there are no incoming edges from $V \setminus S$ to S. Let $|S| = i$. Then the $i \cdot (n - i)$ edges from $V \setminus S$ to S should be absent, and each edge is absent with probability $(1 - p)$. The probability that each vertex in S can reach the target is $R(i, p)$. So the probability of S being the reverse reachable set is given by:

$$(1 - p)^{i \cdot (n-i)} \cdot R(i, p). \tag{2}$$

There are $\binom{n-1}{i-1}$ possible subsets of i vertices that include the given target vertex, and i can range from 1 to n. Exactly one subset S of V will be the reverse reachable set. So the sum of probabilities of the events that S is reverse reachable set is 1. Hence we have:

$$1 = \sum_{i=1}^{n} \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)} \cdot R(i,p) \tag{3}$$

Moving all but the last term (with $i = n$) to the other side, we get the following recurrence relation:

$$R(n,p) = 1 - \sum_{i=1}^{n-1} \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)} \cdot R(i,p). \tag{4}$$

Bound on p for lower bound on $R(n,p)$. We will prove a lower bound on p in terms of n such that the probability that not all n vertices can reach the target vertex is less than $O(1/n)$. In other words, we require

$$R(n,p) \geq 1 - O\left(\frac{1}{n}\right) \tag{5}$$

Since $R(i,p)$ is a probability value, it is at most 1. Hence from Eq. (4) it follows that it suffices to show that

$$\sum_{i=1}^{n-1} \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)} \cdot R(i,p) \leq \sum_{i=1}^{n-1} \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)} \leq O\left(\frac{1}{n}\right) \tag{6}$$

to show that $R(n,p) \geq 1 - O\left(\frac{1}{n}\right)$. We will prove a lower bound on p for achieving Eq. (6). Let us denote by $t_i = \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)}$, for $1 \leq i \leq n-1$. The following lemma establishes a relation of t_i and t_{n-i}.

Lemma 20. *For $1 \leq i \leq n-1$, we have $t_{n-i} = \frac{n-i}{i} \cdot t_i$.*

Proof. We have

$$\begin{aligned} t_{n-i} &= \binom{n-1}{n-i-1}(1-p)^{i\cdot(n-i)} \\ &= \binom{n-1}{i} \cdot (1-p)^{i\cdot(n-i)} \\ &= \frac{n-i}{i} \cdot \binom{n-1}{i-1}(1-p)^{i\cdot(n-i)} \\ &= \frac{n-i}{i} \cdot t_i \end{aligned}$$

The desired result follows. □

Define $g_i = t_i + t_{n-i}$, for $1 \leq i \leq \lfloor n/2 \rfloor$. From the previous lemma we have

$$g_i = t_{n-i} + t_i = \frac{n}{i} \cdot t_i = \frac{n}{i} \cdot \binom{n-1}{i-1} \cdot (1-p)^{i\cdot(n-i)} = \binom{n}{i} \cdot (1-p)^{i\cdot(n-i)}.$$

We now establish a bound on g_i in terms of t_1. In the subsequent lemma we establish a bound on t_1.

Lemma 21. *For sufficiently large n, if $p \geq \frac{c\cdot\log(n)}{n}$ with $c > 2$, then $g_i \leq t_1$ for all $2 \leq i \leq \lfloor \frac{n}{2} \rfloor$.*

Proof. Let $p \geq \frac{c\cdot\log(n)}{n}$ with $c > 2$. Now

$$\begin{aligned} \frac{t_1}{g_i} = \frac{(1-p)^{n-1}}{\binom{n}{i}\cdot(1-p)^{i\cdot(n-i)}} &\geq \frac{1}{n^i \cdot (1-p)^{(i-1)\cdot(n-i-1)}} \quad \text{(Rearranging powers of } (1-p) \text{ and } \binom{n}{i} \leq n^i) \\ &\geq \frac{1}{n^i \cdot e^{\frac{-c\cdot\log(n)}{n}\cdot(i-1)\cdot(n-i-1)}} \quad (1-x \leq e^{-x}) \\ &= n^{\frac{c}{n}\cdot(i-1)\cdot(n-i-1)-i} \end{aligned}$$

To show that $t_1 \geq g_i$, it is sufficient to show that for $2 \leq i \leq \lfloor n/2 \rfloor$,

$$\frac{c}{n} \cdot (i-1) \cdot (n-i-1) - i \geq 0 \Leftrightarrow \frac{i \cdot n}{(i-1)\cdot(n-i-1)} \leq c$$

Note that $f(i) = \frac{i \cdot n}{(i-1)\cdot(n-i-1)}$ is convex for $2 \le i \le \lfloor n/2 \rfloor$. Hence, its maximum value is attained at either of the endpoints. We can see that

$$f(2) = \frac{2 \cdot n}{n-3} \le c \quad \text{(for sufficiently large } n \text{ and } c > 2)$$

and

$$f(\lfloor n/2 \rfloor) = \frac{\lfloor n/2 \rfloor \cdot n}{(\lfloor n/2 \rfloor - 1)\cdot(\lceil n/2 \rceil - 1)}$$

Note that $\lim_{n\to\infty} f(\lfloor n/2 \rfloor) = 2$, and hence for any constant $c > 2$, $f(\lfloor n/2 \rfloor) \le c$ for sufficiently large n. The result follows. □

Lemma 22. *For sufficiently large n, if $p \ge \frac{c\cdot\log(n)}{n}$ with $c > 2$, then $t_1 \le \frac{1}{n^2}$.*

Proof. We have $t_1 = (1-p)^{n-1}$. For $p \ge \frac{c\cdot\log(n)}{n}$ we have

$$t_1 \le \left(1 - \frac{c\cdot\log(n)}{n}\right)^{n-1} \le e^{-\frac{c\cdot\log(n)\cdot(n-1)}{n}} \quad \text{(Since } 1 - x \le e^{-x})$$
$$\le e^{-2\cdot\log(n)} = \frac{1}{n^2} \quad \text{(for sufficiently large } n,\ c > 2)$$

Hence, the desired result follows. □

We are now ready to establish the main lemma that proves the upper bound on $R(n, p)$ and then the main result of the section.

Lemma 23. *For sufficiently large n, for all $p \ge \frac{c\cdot\log(n)}{n}$ with $c > 2$, we have $R(n, p) \ge 1 - \frac{1.5}{n}$.*

Proof. We first show that $\sum_{i=1}^{n-1} t_i \le \frac{1.5}{n}$. We have

$$\begin{aligned}
\sum_{i=1}^{n-1} t_i &= t_1 + t_{n-1} + \sum_{i=2}^{n-2} t_i \\
&\le t_1 + t_{n-1} + \sum_{i=2}^{\lfloor n/2 \rfloor} g_i \quad (t_{\lfloor n/2 \rfloor} \text{ is repeated if } n \text{ is even}) \\
&\le n \cdot t_1 + \sum_{i=2}^{\lfloor n/2 \rfloor} g_i \quad \left(\text{We apply } t_i + t_{n-i} = \frac{n}{i}\cdot t_i \text{ with } i = 1\right) \\
&\le n \cdot t_1 + \sum_{i=2}^{\lfloor n/2 \rfloor} t_1 \quad (\text{By Lemma 21 we have } g_i \le t_1 \text{ for } 2 \le i \le \lfloor n/2 \rfloor) \\
&\le \frac{3\cdot n}{2}\cdot t_1 \\
&\le \frac{3\cdot n}{2\cdot n^2} \quad \left(\text{By Lemma 22 we have } t_1 \le \frac{1}{n^2}\right)
\end{aligned}$$

By Eq. (6) we have that $R(n, p) \ge 1 - \sum_{i=1}^{n-1} t_i$. It follows that $R(n, p) \ge 1 - \frac{1.5}{n}$. □

Theorem 2. *The expected number of iterations of the classical algorithm for MDPs with Büchi objectives for random graphs $\mathcal{G}_{n,p}$, with $p \ge \frac{c\cdot\log(n)}{n}$, where $c > 2$, is $O(1)$, and the average case running time is linear.*

Proof. By Lemma 23 it follows that $R(n, p) \ge 1 - \frac{1.5}{n}$, and if all vertices reach the target set, then the classical algorithm ends in one iteration. In the worst case the number of iterations of the classical algorithm is n. Hence the expected number of iterations is bounded by

$$1 \cdot \left(1 - \frac{1.5}{n}\right) + n \cdot \frac{1.5}{n} = O(1).$$

Since the expected number of iterations is $O(1)$ and every iteration takes linear time, it follows that the average case running time is linear. □

4.2. Average-case analysis over all graphs

In this section, we consider uniform distribution over all graphs, i.e., all possible different graphs are equally likely. This is equivalent to considering the Erdös–Rényi model such that each edge has probability $\frac{1}{2}$. Using $\frac{1}{2} \geq 3 \cdot \log(n)/n$ (for $n \geq 17$) and the results from Section 4.1, we already know that the average case running time for $\mathcal{G}_{n,1/2}$ is linear. In this section we show that in $\mathcal{G}_{n,\frac{1}{2}}$, the probability that not all vertices reach the target is in fact exponentially small in n. It will follow that MDPs where the classical algorithm takes more than constant iterations are exponentially rare. We consider the same recurrence $R(n, p)$ as in the previous subsection and consider t_k and g_k as defined before. The following theorem shows the desired result.

Theorem 3. *In $\mathcal{G}_{n,\frac{1}{2}}$ with sufficiently large n the probability that the classical algorithm takes more than one iteration is less than $\left(\frac{3}{4}\right)^n$.*

Proof. We first observe that Eq. (4) and Eq. (6) holds for all probabilities. Next we observe that Lemma 21 holds for $p \geq \frac{c \cdot \log(n)}{n}$ with any constant $c > 2$, and hence also for $p = \frac{1}{2}$ for sufficiently large n. Hence by applying the inequalities of the proof of Lemma 23 we obtain that

$$\sum_{i=1}^{n-1} t_i \leq \frac{3 \cdot n}{2} \cdot t_1 .$$

For $p = \frac{1}{2}$ we have $t_1 = \binom{n-1}{0} \cdot \left(1 - \frac{1}{2}\right)^{n-1} = \frac{1}{2^{n-1}}$. Hence we have

$$R(n, p) \geq 1 - \frac{3 \cdot n}{2 \cdot 2^{n-1}} > 1 - \frac{1.5^n}{2^n} = 1 - \left(\frac{3}{4}\right)^n .$$

The second inequality holds for sufficiently large n. It follows that the probability that the classical algorithm takes more than one iteration is less than $(\frac{3}{4})^n$. The desired result follows. □

Appendix A. Technical appendix

Proposition 1 *(Useful inequalities from Stirling inequalities). For natural numbers ℓ and j with $j \leq \ell$ we have the following inequalities:*

1. $\binom{\ell}{j} \leq \left(\frac{e \cdot \ell}{j}\right)^j$.
2. $\binom{\ell}{j} \leq (\ell + 1) \cdot \left(\frac{\ell}{j}\right)^j \cdot \left(\frac{\ell}{\ell - j}\right)^{\ell - j}$.

Proof. The proof of the results is based on the following Stirling inequality for factorial:

$$e \cdot \left(\frac{j}{e}\right)^j \leq j! \leq e \cdot \left(\frac{j+1}{e}\right)^{j+1} .$$

We now use the inequality to show the desired inequalities:

1. We have

$$\binom{\ell}{j} \leq \frac{\ell^j}{j!} \leq \frac{\ell^j \cdot e^j}{e \cdot j^j} \quad \text{(using Stirling inequality)}$$
$$\leq \frac{1}{e} \cdot \left(\frac{e \cdot \ell}{j}\right)^j$$
$$\leq \left(\frac{e \cdot \ell}{j}\right)^j$$

2. We have

$$\begin{aligned}
\binom{\ell}{j} &= \frac{\ell!}{j!\cdot(n-j)!} \\
&\le \left(e\cdot\left(\frac{\ell+1}{e}\right)^{\ell+1}\right)\cdot\left(\frac{1}{e\cdot\left(\frac{j}{e}\right)^{j}\cdot e\cdot\left(\frac{\ell-j}{e}\right)^{\ell-j}}\right) \\
&= \frac{1}{e^2}\cdot(\ell+1)\cdot\left(\frac{\ell+1}{j}\right)^{j}\cdot\left(\frac{\ell+1}{\ell-j}\right)^{\ell-j} \\
&\le \frac{1}{e^2}\cdot(\ell+1)\cdot\left(\frac{\ell+1}{\ell}\right)^{\ell}\left(\frac{\ell}{j}\right)^{j}\cdot\left(\frac{\ell}{\ell-j}\right)^{\ell-j} \\
&\le \frac{1}{e^2}\cdot(\ell+1)\cdot e\cdot\left(\frac{\ell}{j}\right)^{j}\cdot\left(\frac{\ell}{\ell-j}\right)^{\ell-j} \qquad \left(\text{Since } \left(1+\frac{1}{\ell}\right)^{\ell}\le e\right) \\
&\le (\ell+1)\cdot\left(\frac{\ell}{j}\right)^{j}\cdot\left(\frac{\ell}{\ell-j}\right)^{\ell-j}
\end{aligned}$$

The first inequality is obtained by applying the Stirling inequality to the numerator (in the first term), and applying the Stirling inequality twice to the denominator (in the second term). □

References

[1] C. Baier, J.-P. Katoen, Principles of Model Checking, MIT Press, 2008.
[2] A. Bianco, L. de Alfaro, Model checking of probabilistic and nondeterministic systems, in: FSTTCS, in: LNCS, vol. 1026, Springer, 1995, pp. 499–513.
[3] K. Chatterjee, Stochastic ω-regular games, PhD thesis, UC Berkeley, 2007.
[4] K. Chatterjee, The complexity of stochastic Müller games, Inform. and Comput. 211 (2012) 29–48.
[5] K. Chatterjee, L. de Alfaro, M. Faella, R. Majumdar, V. Raman, Code-aware resource management, Form. Methods Syst. Des. 42 (2) (2013) 146–174.
[6] K. Chatterjee, L. de Alfaro, T.A. Henzinger, The complexity of stochastic Rabin and Streett games, in: ICALP, in: LNCS, vol. 3580, Springer, 2005, pp. 878–890.
[7] K. Chatterjee, M. Henzinger, Faster and dynamic algorithms for maximal end-component decomposition and related graph problems in probabilistic verification, in: SODA, ACM–SIAM, 2011, pp. 1318–1336.
[8] K. Chatterjee, M. Henzinger, An $O(n^2)$ algorithm for alternating Büchi games, in: SODA, ACM–SIAM, 2012, pp. 1386–1399.
[9] K. Chatterjee, M. Henzinger, Efficient and dynamic algorithms for alternating Büchi games and maximal end-component decomposition, J. ACM 61 (3) (2014).
[10] K. Chatterjee, M. Henzinger, M. Joglekar, N. Shah, Symbolic algorithms for qualitative analysis of Markov decision processes with Büchi objectives, Form. Methods Syst. Des. 42 (3) (2013) 301–327.
[11] K. Chatterjee, M. Jurdziński, T.A. Henzinger, Simple stochastic parity games, in: CSL, in: LNCS, vol. 2803, Springer, 2003, pp. 100–113.
[12] K. Chatterjee, M. Jurdziński, T.A. Henzinger, Quantitative stochastic parity games, in: SODA, SIAM, 2004, pp. 121–130.
[13] K. Chatterjee, J. Lacki, Faster algorithms for Markov decision processes with low treewidth, in: CAV, in: LNCS, vol. 8044, Springer, 2013, pp. 543–558.
[14] C. Courcoubetis, M. Yannakakis, The complexity of probabilistic verification, J. ACM 42 (4) (1995) 857–907.
[15] L. de Alfaro, Formal verification of probabilistic systems, PhD thesis, Stanford University, 1997.
[16] L. de Alfaro, P. Roy, Magnifying-lens abstraction for Markov decision processes, in: CAV, in: LNCS, vol. 4590, Springer, 2007, pp. 325–338.
[17] P. Erdös, A. Rényi, On the evolution of random graphs, Publ. Math. Inst. Hung. Acad. Sci. 5 (1960) 17–61.
[18] J. Filar, K. Vrieze, Competitive Markov Decision Processes, Springer-Verlag, 1997.
[19] H. Howard, Dynamic Programming and Markov Processes, MIT Press, 1960.
[20] M. Kwiatkowska, G. Norman, D. Parker, Verifying randomized distributed algorithms with PRISM, in: Workshop on Advances in Verification, WAVe, 2000.
[21] A. Pogosyants, R. Segala, N. Lynch, Verification of the randomized consensus algorithm of Aspnes and Herlihy: a case study, Distrib. Comput. 13 (3) (2000) 155–186.
[22] M.L. Puterman, Markov Decision Processes, J. Wiley and Sons, 1994.
[23] R. Segala, Modeling and verification of randomized distributed real-time systems, PhD thesis, MIT, 1995, Technical report MIT/LCS/TR-676.
[24] M.I.A. Stoelinga, Fun with FireWire: experiments with verifying the IEEE1394 root contention protocol, Form. Asp. Comput. 14 (3) (2002) 328–337.
[25] W. Thomas, Languages, automata, and logic, in: G. Rozenberg, A. Salomaa (Eds.), Beyond Words, in: Handbook of Formal Languages, vol. 3, Springer, 1997, pp. 389–455, Chapter 7.
[26] M. Kwiatkowska, G. Norman, D. Parker, PRISM: probabilistic symbolic model checker, in: TOOLS, in: LNCS, vol. 2324, Springer, 2002, pp. 200–204.