



# Solving Vertex Cover in Polynomial Time on Hyperbolic Random Graphs

Thomas Bläsius<sup>1</sup> · Philipp Fischbeck<sup>2</sup> · Tobias Friedrich<sup>2</sup> · Maximilian Katzmann<sup>3</sup>

Accepted: 13 September 2021 / Published online: 28 October 2021  
© The Author(s) 2021

## Abstract

The computational complexity of the VERTEXCOVER problem has been studied extensively. Most notably, it is NP-complete to find an optimal solution and typically NP-hard to find an approximation with reasonable factors. In contrast, recent experiments suggest that on many real-world networks the run time to solve VERTEXCOVER is way smaller than even the best known FPT-approaches can explain. We link these observations to two properties that are observed in many real-world networks, namely a heterogeneous degree distribution and high clustering. To formalize these properties and explain the observed behavior, we analyze how a branch-and-reduce algorithm performs on hyperbolic random graphs, which have become increasingly popular for modeling real-world networks. In fact, we are able to show that the VERTEXCOVER problem on hyperbolic random graphs can be solved in polynomial time, with high probability. The proof relies on interesting structural properties of hyperbolic random graphs. Since these predictions of the model are interesting in their own right, we conducted experiments on real-world networks showing that these properties are also observed in practice.

**Keywords** Vertex cover · Random graphs · Hyperbolic geometry · Efficient algorithm

---

This article belongs to the Topical Collection: *Special Issue on Theoretical Aspects of Computer Science (STACS 2020)*

Guest Editors: Christophe Paul and Markus Bläser

A preliminary version of this paper appeared in [6]

✉ Maximilian Katzmann  
maximilian.katzmann@hpi.de

Extended author information available on the last page of the article.

## 1 Introduction

The VERTEXCOVER problem is one of the most fundamental NP-complete graph problems. Given an undirected graph  $G$  on  $n$  vertices the goal is to find the smallest vertex subset  $S$ , such that each edge in  $G$  is incident to at least one vertex in  $S$ . Since, by definition, there can be no edge between two vertices outside of  $S$ , these remaining vertices form an independent set. Therefore, one can easily derive a maximal independent set from a minimal vertex cover and vice versa.

Due to its NP-completeness there is probably no polynomial time algorithm for solving VERTEXCOVER. The best known algorithm for INDEPENDENTSET runs in  $1.1996^n \text{poly}(n)$  [26]. To analyze the complexity of VERTEXCOVER on a finer scale, several parameterized solutions have been proposed. One can determine whether a graph  $G$  has a vertex cover of size  $k$  by applying a *branch-and-reduce* algorithm. The idea is to build a search tree by recursively considering two possible extensions of the current vertex cover (*branching*), until a vertex cover is found or the size of the current cover exceeds  $k$ . Each branching step is followed by a *reduce* step in which *reduction rules* are applied to make the considered graph smaller. This branch-and-reduce technique yields a simple  $\mathcal{O}(2^k \text{poly}(n))$  algorithm, where the exponential portion comes from the branching. The best known FPT (fixed-parameter tractable) algorithm runs in  $\mathcal{O}(1.2738^k + kn)$  time [12], and unless ETH (exponential time hypothesis) fails, there can be no  $2^{o(k)} \text{poly}(n)$  algorithm [11].

While these FPT approaches promise relatively small running times if the considered network has a small vertex cover, the optimal solution is large for many real-world networks. Nevertheless, it was recently observed that applying a branch-and-reduce technique on real instances is very efficient [1]. Some of the considered networks had millions of vertices, yet an optimal solution (also containing millions of vertices) was computed within seconds. Most instances were solved so quickly since the expensive branching was not necessary at all. In fact, the application of the reduction rules alone already yielded an optimal solution. Most notably, applying the *dominance reduction rule*, which eliminates vertices whose neighborhood contains a vertex together with its neighborhood, reduces the graph to a very small remainder on which the branching, if necessary, can be done quickly. We trace the effectiveness of the dominance rule back to two properties that are often observed in real-world networks: a *heterogeneous degree distribution* (the network contains many vertices of small degree and few vertices of high degree) and *high clustering* (the neighbors of a vertex are likely to be neighbors themselves).

We formalize these key properties using *hyperbolic random graphs* to analyze the performance of the dominance rule. Introduced by Krioukov et al. [20], hyperbolic random graphs are obtained by randomly distributing vertices in the hyperbolic plane and connecting any two that are geometrically close. The resulting graphs feature a power-law degree distribution and high clustering [18, 20] (the two desired properties), which can be tuned using parameters of the model. Additionally, the generated networks have a small diameter [17, 19]. All of these properties have been observed in many real-world networks such as the internet, social networks, as well as biological networks like protein-protein interaction networks [2, 3, 14]. Furthermore, Boguná, Papadopoulos, and Krioukov showed that the internet can be embedded into

the hyperbolic plane such that routing packages between network participants, greedily with respect to the hyperbolic distance, leads to routes that are very close to the shortest paths in the graph [10]. This correlation between hyperbolic distances and path lengths gives reason to believe that the network fits naturally into the hyperbolic plane.

Recently it has been shown that on hyperbolic random graphs VERTEXCOVER can be approximated in quasi-linear time within a factor of  $1 + o(1)$ , asymptotically almost surely [8]. Here, we extend this work by showing that VERTEXCOVER can be solved *exactly* in polynomial time on hyperbolic random graphs, with high probability. This is done by proving that even a single application of the dominance reduction rule reduces a hyperbolic random graph to a remainder with small pathwidth on which VERTEXCOVER can then be solved efficiently. Our analysis provides an explanation for why VERTEXCOVER can be solved efficiently on practical instances. We note that, while our analysis makes use of the underlying hyperbolic geometry, the algorithm itself is oblivious to it. Since our proof relies on certain structural properties of hyperbolic random graphs, we conducted experiments to test whether these are also found in real-world networks. Our results indicate that these predictions actually match the real world for a significant fraction of networks.

## 2 Preliminaries

Let  $G = (V, E)$  be an undirected graph. We denote the number of vertices in  $G$  with  $n$ . The *neighborhood* of a vertex  $v$  is defined as  $N(v) = \{w \in V \mid \{v, w\} \in E\}$  and the size of the neighborhood, called the *degree* of  $v$ , is denoted by  $\deg(v)$ . For a subset  $S \subseteq V$ , we use  $G[S]$  to denote the induced subgraph of  $G$  obtained by removing all vertices in  $V \setminus S$ .

**The Hyperbolic Plane** The hyperbolic plane  $\mathbb{H}^2$  is an infinite two-dimensional surface of constant negative curvature. For a detailed introduction to hyperbolic geometry, we refer the reader to the book by Ramsay and Richtmyer [23]. There are several models that can be used to represent  $\mathbb{H}^2$  (see [23, Chapter 7.8]). In this paper, we use the *native representation* (also called *polar-coordinate model*) of the hyperbolic plane, which is defined as follows. After choosing a designated pole  $O \in \mathbb{H}^2$ , together with a polar axis, i.e., a reference ray starting at  $O$ , a point  $p$  is uniquely identified by its *radius*  $r(p)$ , denoting the hyperbolic distance to  $O$ , and its *angle* (or *angular coordinate*)  $\varphi(p)$ , denoting the angular distance between the polar axis and the line through  $p$  and  $O$ . The hyperbolic distance between two points  $p$  and  $q$  is given by

$$\text{dist}(p, q) = \text{acosh}(\cosh(r(p)) \cosh(r(q)) - \sinh(r(p)) \sinh(r(q)) \cos(\Delta_\varphi(p, q))),$$

where  $\cosh(x) = (e^x + e^{-x})/2$ ,  $\sinh(x) = (e^x - e^{-x})/2$  (both growing as  $e^x/2 \pm o(1)$ ), and  $\Delta_\varphi(p, q) = \pi - |\pi - |\varphi(p) - \varphi(q)||$  denotes the angular distance between  $p$  and  $q$ . If not stated otherwise, we assume that computations on angles are performed modulo  $2\pi$ .

We use  $B_p(r)$  to denote a disk of radius  $r$  centered at  $p$ , i.e., the set of points with hyperbolic distance at most  $r$  to  $p$ . Such a disk has an area of  $2\pi(\cosh(r) -$

1) and circumference  $2\pi \sinh(r)$ . Thus, the area and the circumference of a disk in the hyperbolic plane grow exponentially with its radius. In contrast, this growth is polynomial in Euclidean space. Therefore, representing hyperbolic shapes in the Euclidean geometry results in a distortion. In the *native representation*, used in our figures, circles can appear teardrop-shaped (see Fig. 2).

**Hyperbolic Random Graphs** Hyperbolic random graphs are obtained by distributing  $n$  points uniformly at random within the disk  $B_O(R)$ , as explained below, and connecting any two of them if and only if their hyperbolic distance is at most  $R$ ; see Fig. 1. The disk radius  $R$  (which matches the connection threshold) depends on  $n$ , as well as the power-law exponent  $\beta = 2\alpha + 1$  (for  $\alpha \in (1/2, 1)$ ) and the average degree  $\kappa$  of the generated network, both of which are assumed to be constant. More precisely,  $R$  is given by

$$R = 2 \log \left( \frac{2n}{\pi\kappa} \left( \frac{\alpha}{\alpha - 1/2} \right)^2 (1 + o(1)) \right). \quad ([18, Theorem 23])$$

The coordinates for the vertices are drawn as follows. For vertex  $v$  the angular coordinate, denoted by  $\varphi(v)$ , is drawn uniformly at random from  $[0, 2\pi)$ . The radius of  $v$ , denoted by  $r(v)$ , is sampled according to the probability density function

$$f(r) = \frac{\alpha \sinh(\alpha r)}{\cosh(\alpha R) - 1} = \alpha e^{-\alpha(R-r)} (1 + \Theta(e^{-\alpha R} - e^{-2\alpha r}))$$

for  $r \in [0, R]$ . For  $r > R$ ,  $f(r) = 0$ . This function grows exponentially as  $r$  approaches  $R$ . The joint distribution function of angles and radii is then given by

$$f(r, \varphi) = \frac{1}{2\pi} f(r). \quad (1)$$

Note that we obtain power-law exponents  $\beta \in (2, 3)$ . Exponents outside of this range are atypical for hyperbolic random graphs. On the one hand, for  $\beta < 2$  the average degree of the generated networks is divergent. On the other hand, for  $\beta > 3$  hyperbolic random graphs degenerate: They decompose into smaller components, none having a size linear in  $n$ . The obtained graphs have logarithmic tree width [9], meaning the VERTEXCOVER problem can be solved efficiently in that case.

The probability for a given vertex to lie in a certain area  $A$  of the disk is given by its probability measure  $\mu(A) = \iint_A f(r, \varphi) d\varphi dr$ . The hyperbolic distance between two vertices  $u$  and  $v$  increases with increasing angular distance between them. The maximum angular distance such that they are still connected by an edge is bounded by [18, Lemma 6]

$$\begin{aligned} \theta(r(u), r(v)) &= \arccos \left( \frac{\cosh(r(u)) \cosh(r(v)) - \cosh(R)}{\sinh(r(u)) \sinh(r(v))} \right) \\ &= 2e^{(R-r(u)-r(v))/2} (1 + \Theta(e^{R-r(u)-r(v)})). \end{aligned} \quad (2)$$

**Interval Graphs and Circular Arc Graphs** In an interval graph each vertex  $v$  is identified with an interval on the real line and two vertices are adjacent if and only if their intervals intersect. The *interval width* of an interval graph  $G$ , denoted by  $iw(G)$ , is its maximum clique size, i.e., the maximum number of intervals that intersect in one

point. For any graph the interval width is defined as the minimum interval width over all of its interval supergraphs. Circular arc graphs are a superclass of interval graphs, where each vertex is identified with a subinterval of the circle called *circular arc* or simply *arc*. The interval width of a circular arc graph  $G$  is at most twice the size of its maximum clique, since one obtains an interval supergraph of  $G$  by mapping the circular arcs into the interval  $[0, 2\pi]$  on the real line and replacing all intervals that were split by this mapping with the whole interval  $[0, 2\pi]$ . Consequently, for any graph  $G$ , if  $k$  denotes the minimum over the maximum clique number of all circular arc supergraphs  $G'$  of  $G$ , then the interval width of  $G$  is at most  $2k$ .

**Treewidth and Pathwidth** A *tree decomposition* of a graph  $G$  is a tree  $T$  where each tree node represents a subset of the vertices of  $G$  called a *bag*, and the following requirements have to be satisfied: Each vertex in  $G$  is contained in at least one bag, all bags containing a given vertex in  $G$  form a connected subtree of  $T$ , and for each edge in  $G$ , there exists a bag containing both endpoints. The *width* of a tree decomposition is the size of its largest bag minus one. The *treewidth* of  $G$  is the minimum width over all tree decompositions of  $G$ . The *path decomposition* of a graph is defined analogously to the tree decomposition, with the constraint that the tree has to be a path. Additionally, as for the treewidth, the *pathwidth* of a graph  $G$ , denoted by  $\text{pw}(G)$ , is the minimum width over all path decompositions of  $G$ . Clearly the pathwidth is an upper bound on the treewidth. It is known that for any graph  $G$  and any  $k \geq 0$ , the interval width of  $G$  is at most  $k + 1$  if and only if its pathwidth is at most  $k$  [13, Theorem 7.14]. Consequently, if  $k'$  is the maximum clique size of a circular arc supergraph of  $G$ , then  $2k' - 1$  is an upper bound on the pathwidth of  $G$ .

**Probabilities** Since we are analyzing a random graph model, our results are of probabilistic nature. To obtain meaningful statements, we show that they hold *with high probability*, i.e., with probability  $1 - \mathcal{O}(n^{-1})$ . The following Chernoff bounds consider the probability for a random variable to deviate too much from its expected value. This is a useful tool for showing that certain events occur with high probability.

**Theorem 1** (Chernoff Bound [15, Theorem 1.1]) *Let  $X_1, \dots, X_n$  be independent random variables with  $X_i \in \{0, 1\}$  and let  $X$  be their sum. Then, for  $\varepsilon \in (0, 1)$*

$$\begin{aligned}\Pr[X > (1 + \varepsilon)\mathbb{E}[X]] &\leq e^{-\varepsilon^2/3 \cdot \mathbb{E}[X]}, \\ \Pr[X < (1 - \varepsilon)\mathbb{E}[X]] &\leq e^{-\varepsilon^2/2 \cdot \mathbb{E}[X]}.\end{aligned}$$

Usually, it is sufficient to show that a random variable does not exceed a certain upper bound, with high probability. The following corollary shows that an upper bound on the expected value suffices to obtain concentration.

**Corollary 1** *Let  $X_1, \dots, X_n$  be independent random variables with  $X_i \in \{0, 1\}$ , let  $X$  be their sum, and let  $f(n)$  be an upper bound on  $\mathbb{E}[X]$ . Then, for all  $\varepsilon \in (0, 1)$  it holds that*

$$\Pr[X > (1 + \varepsilon)f(n)] \leq e^{-\varepsilon^2/3 \cdot f(n)}.$$

*Proof* Consider a random variable  $X'$  with  $f(n) = \mathbb{E}[X']$  such that  $X \leq X'$  for every outcome. Note that  $X'$  exists as  $f(n) \geq \mathbb{E}[X]$ . Since  $X \leq X'$ , it holds that

$$\Pr[X > (1 + \varepsilon)f(n)] \leq \Pr[X' > (1 + \varepsilon)f(n)] = \Pr[X' > (1 + \varepsilon)\mathbb{E}[X']].$$

Using Theorem 1 we can derive that

$$\Pr[X' > (1 + \varepsilon)\mathbb{E}[X']] \leq e^{-\varepsilon^2/3 \cdot \mathbb{E}[X']} = e^{-\varepsilon^2/3 \cdot f(n)}.$$

□

### 3 Vertex Cover on Hyperbolic Random Graphs

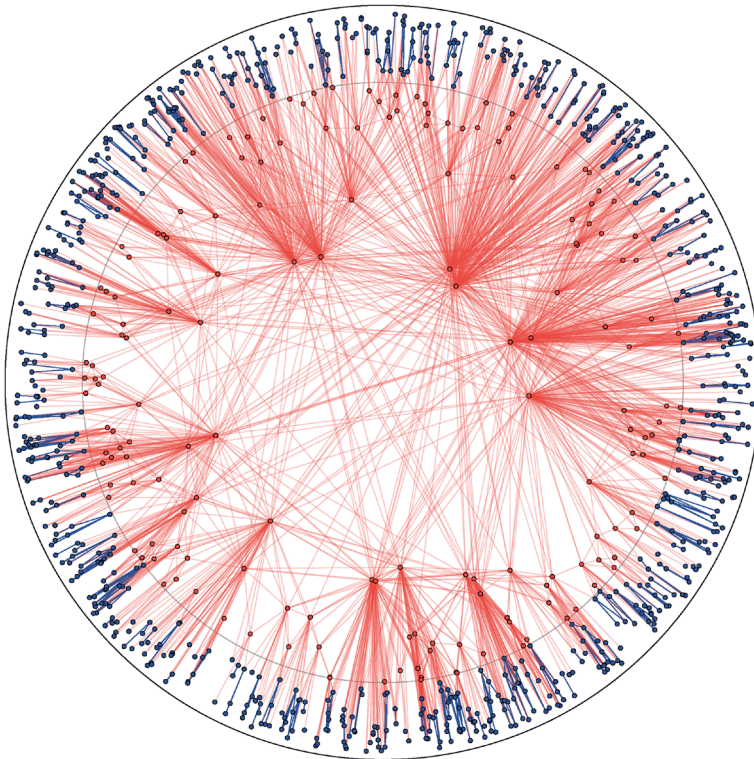
Reduction rules are often applied as a preprocessing step, before using a brute force search or branching in a search tree. They simplify the input by removing parts that are easy to solve. For example, an isolated vertex does not cover any edges and can thus never be part of a minimum vertex cover. Consequently, in a preprocessing step all isolated vertices can be removed, which leads to a reduced input size without impeding the search for a minimum.

The dominance reduction rule was previously defined for the INDEPENDENTSET problem [16], and later used for VERTEXCOVER in the experiments by Akiba and Iwata [1]. Formally, vertex  $u$  *dominates* a neighbor  $v \in N(u)$  if  $(N(v) \setminus \{u\}) \subseteq N(u)$ , i.e., all neighbors of  $v$  are also neighbors of  $u$ . We say  $u$  is *dominant* if it dominates at least one vertex. The dominance rule states that  $u$  can be added to the vertex cover (and afterwards removed from the graph), without impeding the search for a minimum vertex cover. To see that this is correct, assume that  $u$  dominates  $v$  and let  $S$  be a minimum vertex cover that does not contain  $u$ . Since  $S$  has to cover all edges, it contains all neighbors of  $u$ . These neighbors include  $v$  and all of  $v$ 's neighbors, since  $u$  dominates  $v$ . Therefore, removing  $v$  from  $S$  leaves only the edge  $\{u, v\}$  uncovered which can be fixed by adding  $u$  instead. The resulting vertex cover has the same size as  $S$ . When searching for a minimum vertex cover of  $G$ , it is thus safe to assume that  $u$  is part of the solution and to reduce the search to  $G[V \setminus \{u\}]$ .

In the remainder of this section, we study the effectiveness of the dominance reduction rule on hyperbolic random graphs and conclude that VERTEXCOVER can be solved efficiently on these graphs. Our results are summarized in the following main theorem.

**Theorem 2** *Let  $G$  be a hyperbolic random graph on  $n$  vertices. Then the VERTEXCOVER problem on  $G$  can be solved in  $\text{poly}(n)$  time, with high probability.*

The proof of Theorem 2 consists of two parts that make use of the underlying hyperbolic geometry. In the first part, we show that applying the dominance reduction rule once removes all vertices in the inner part of the hyperbolic disk with high probability, as depicted in Fig. 1. We note that this is independent of the order in which the reduction rule is applied, as dominant vertices remain dominant after removing other dominant vertices. In the second part, we consider the induced subgraph containing the remaining vertices near the boundary of the disk (blue vertices in Fig. 1). We prove that this subgraph has a small pathwidth, by showing that there is a circular

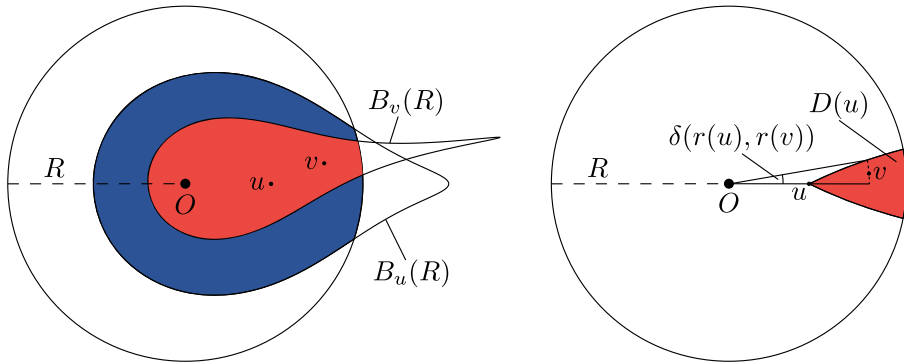


**Fig. 1** A hyperbolic random graph with 979 vertices, average degree 8.3, and a power-law exponent of 2.5. In such a graph the red vertices and edges are removed by the dominance reduction rule, with high probability. Additionally, the remaining subgraph in the outer band (consisting of the blue vertices and edges) has a small path width, with high probability

arc supergraph with a small interval width. Consequently, a tree decomposition of this subgraph can be computed efficiently. Finally, we obtain a polynomial time algorithm for VERTEXCOVER by first applying the reduction rules and afterwards solving VERTEXCOVER on the remaining subgraph using dynamic programming on the tree decomposition of small width.

### 3.1 Dominance on Hyperbolic Random Graphs

Recall that a hyperbolic random graph is obtained by distributing  $n$  vertices in a hyperbolic disk  $B_O(R)$  and that any two are connected if their distance is at most  $R$ . Consequently, one can imagine the neighborhood of a vertex  $u$  as another disk  $B_u(R)$ . Vertex  $u$  dominates another vertex  $v$  if its neighborhood disk completely contains that of  $v$  (both constrained to  $B_O(R)$ ), as depicted in Fig. 2 (left). We define the *dominance area*  $D(u)$  of  $u$  to be the area containing all such vertices  $v$ . That is,  $D(u) = \{p \in B_O(R) \mid B_p(R) \cap B_O(R) \subseteq B_u(R)\}$ . The result is illustrated in Fig. 2 (right). We note that it is sufficient for a vertex  $v$  to lie in  $D(u)$  in order to be dominated by  $u$ , however, it is not necessary.



**Fig. 2** Left: Vertex  $u$  dominates vertex  $v$ , as  $B_v(R) \cap B_O(R)$  (red) is completely contained in  $B_u(R) \cap B_O(R)$  (red and blue). Right: All vertices that lie in  $D(u)$  (red) are dominated by  $u$

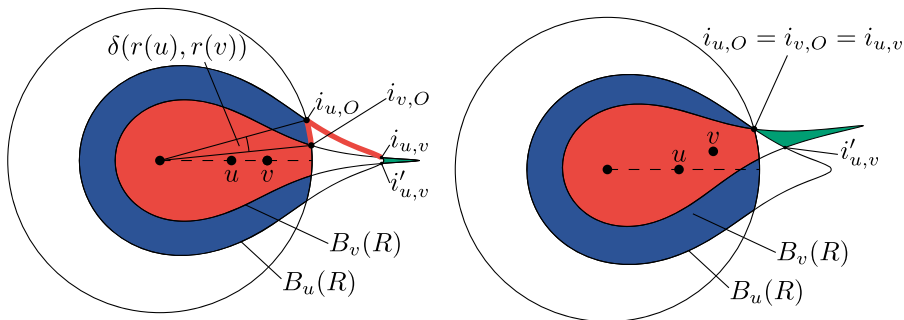
Given the radius  $r(u)$  of vertex  $u$  we can now compute a lower bound on the probability that  $u$  dominates another vertex, i.e., the probability that at least one vertex lies in  $D(u)$ , by determining the measure  $\mu(D(u))$ . To this end, we first define  $\delta(r(u), r(v))$  to be the maximum angular distance between two vertices  $u$  and  $v$  such that  $v$  lies in  $D(u)$ .

**Lemma 1** *Let  $u, v \in B_O(R)$  be two points. Then,  $v \in D(u)$  if and only if  $r(v) \geq r(u)$  and  $\Delta_\varphi(u, v) \leq \delta(r(u), r(v))$ , where*

$$\delta(r(u), r(v)) = 2(e^{-r(u)/2} - e^{-r(v)/2}) + \Theta(e^{-3/2 \cdot r(u)}) - \Theta(e^{-3/2 \cdot r(v)}).$$

*Proof* To prove the claim, we consider the possible positions that  $v$  can have relative to  $u$  and identify the ones for which  $v \in D(u)$  holds.

Assume without loss of generality that  $\varphi(u) = 0$ , as depicted in Fig. 3. By definition,  $v \in D(u)$  if and only if  $B_v(R) \cap B_O(R) \subseteq B_u(R)$ . First note that this is



**Fig. 3** Left: Vertex  $v$  is in the dominance area of  $u$ , since  $B_v(R) \cap B_O(R)$  (red area) is contained in  $B_u(R)$ . The intersections  $i_{u,v}, i'_{u,v}$  mark the separation between  $B_v(R) \setminus B_u(R)$  (green area) and the rest of  $B_v(R)$ . If  $v$  is rotated in counterclockwise direction,  $i_{v,O}$  and  $i_{u,v}$  move along the red lines towards  $i_{u,O}$ . Right: Vertex  $v$  is rotated such that  $i_{u,v} = i_{u,O}$



not the case if  $r(v) < r(u)$ , as then for the point  $p = (R - r(v), \pi)$  it holds that  $p \in B_v(R) \cap B_O(R)$  but  $p \notin B_u(R)$  for all  $\varphi(v) \in [0, 2\pi)$ . For the case when  $r(v) \geq r(u)$ , it was shown that  $B_v(R) \cap B_O(R) \subseteq B_u(R)$  holds when  $u$  and  $v$  have the same angular coordinate [7, Lemma 1]. This shows that the first condition ( $r(v) \geq r(u)$ ) is necessary for  $v$  to be in the dominance area of  $u$ , and it remains to determine the maximum angular deviation between the two points, such that this is still the case.

To this end, we argue about intersections of  $B_u(R)$ ,  $B_v(R)$ , and  $B_O(R)$ , which we use as indicators whether  $v \in D(u)$  holds. For now assume that  $\varphi(v) = \varphi(u)$  and consider the two intersections  $i_{u,v}, i'_{u,v}$  of  $B_u(R)$  with  $B_v(R)$ , as depicted in Fig. 3 (left). Since  $B_v(R) \cap B_O(R) \subseteq B_u(R)$  holds by [7, Lemma 1] and since circles are convex, we know that  $B_v(R) \setminus B_u(R)$  (the green area in Fig. 3 (left)) lies outside of  $B_O(R)$  and so do the two intersections  $i_{u,v}, i'_{u,v}$ . For the same reason, we know that  $i_{v,O}$ , the intersection of  $B_v(R)$  with  $B_O(R)$  with  $\varphi(i_{v,O}) \in [0, \pi]$ , lies in  $B_u(R)$ . It follows that, for the analogously defined intersection  $i_{u,O}$  we have  $\varphi(i_{v,O}) \leq \varphi(i_{u,O})$ .

We now relax the assumption that  $\varphi(v) = \varphi(u)$  and instead imagine that we increase the angle between  $u$  and  $v$  by some  $\delta > 0$ , which denotes a counterclockwise rotation of  $v$  around the origin. (For symmetry reasons the argumentation about a clockwise rotation is analogous.) Then,  $i_{u,v}$  and  $i'_{u,v}$  move along the boundary of  $B_u(R)$  and, in particular,  $i_{u,v}$  moves towards  $i_{u,O}$ . Note that at the same time  $i_{v,O}$  moves towards  $i_{u,O}$  as well. Both movements are depicted using red lines in Fig. 3 (left). As long as  $i_{u,v}$  has not surpassed  $i_{u,O}$ , neither of the two intersections of  $B_v(R)$  with  $B_u(R)$  lies inside of  $B_O(R)$ , which means that  $B_v(R) \setminus B_u(R)$  remains outside of  $B_O(R)$  and we maintain the property that  $B_v(R) \cap B_O(R) \subseteq B_u(R)$ . As we keep increasing  $\delta$ , we eventually get to the point where  $i_{u,v}$  reaches  $i_{u,O}$ , as depicted in Fig. 3 (right). Note that at this point we also have  $i_{v,O} = i_{u,v}$ . Consequently, if we were to rotate  $v$  any further, we would have  $i_{v,O} \notin B_u(R)$ , meaning  $B_v(R) \cap B_O(R)$  would no longer be a subset of  $B_u(R)$ . It follows that  $B_v(R) \cap B_O(R) \subseteq B_u(R)$  if and only if  $\varphi(i_{v,O}) \leq \varphi(i_{u,O})$ .

To compute the maximum angular distance between  $u$  and  $v$  such that this is the case, we again start with the assumption that  $\varphi(v) = \varphi(u) = 0$ , and determine the maximum angle  $\delta(r(u), r(v))$  such that  $\varphi(i_{v,O}) + \delta(r(u), r(v)) \leq \varphi(i_{u,O})$ . Since  $i_{u,O}$  and  $i_{v,O}$  have radius  $R$  and hyperbolic distance  $R$  from  $u$  and  $v$ , respectively, we can apply (2) to compute their angular coordinates as  $\varphi(i_{u,O}) = \theta(r(u), R)$  and  $\varphi(i_{v,O}) = \theta(r(v), R)$ , respectively. Substituting these angles in the above inequality yields  $\theta(r(v), R) + \delta(r(u), r(v)) \leq \theta(r(u), R)$ . We can now solve for  $\delta(r(u), r(v))$  and apply (2) to obtain

$$\begin{aligned} \delta(r(u), r(v)) &= \theta(r(u), R) - \theta(r(v), R) \\ &= 2(e^{-r(u)/2} - e^{-r(v)/2}) + \Theta(e^{-3/2 \cdot r(u)}) - \Theta(e^{-3/2 \cdot r(v)}). \end{aligned}$$

□

Using Lemma 1 we can now compute the probability for a given vertex to lie in the dominance area of  $u$ . We note that this probability grows roughly like  $2/\pi \cdot e^{-r(u)/2}$ ,

which is a constant fraction of the measure of the neighborhood disk of  $u$  which grows as  $\alpha/(\alpha - 1/2) \cdot 2/\pi \cdot e^{-r(u)/2}$  [18, Lemma 3.2]. Consequently, the expected number of vertices that  $u$  dominates at least is a constant fraction of the expected number of its neighbors.

**Lemma 2** *Let  $u$  be a vertex with radius  $r(u) \geq R/2$ . The probability for a given vertex to lie in  $D(u)$  is given by*

$$\mu(D(u)) = \frac{2}{\pi} e^{-r(u)/2} (1 - \mathcal{O}(e^{-\alpha(R-r(u))})) \pm \mathcal{O}(1/n).$$

*Proof* The probability for a given vertex  $v$  to lie in  $D(u)$  is obtained by integrating the probability density (given by (1)) over  $D(u)$ .

$$\begin{aligned} \mu(D(u)) &= 2 \int_{r(u)}^R \int_0^{\delta(r(u),r)} f(r, \varphi) \, d\varphi \, dr \\ &= 2 \int_{r(u)}^R \left( 2(e^{-r(u)/2} - e^{-r/2}) + \Theta(e^{-3/2 \cdot r(u)}) - \Theta(e^{-3/2 \cdot r}) \right) \\ &\quad \cdot \frac{\alpha}{2\pi} e^{-\alpha(R-r)} (1 + \Theta(e^{-\alpha R} - e^{-2\alpha r})) \, dr \end{aligned}$$

Since  $r(u) \geq R/2$  and  $r \in [r(u), R]$  we have  $\Theta(e^{-3/2 \cdot r(u)}) - \Theta(e^{-3/2 \cdot r}) = \pm \mathcal{O}(e^{-3/4 \cdot R})$  and  $(1 + \Theta(e^{-\alpha R} - e^{-2\alpha r})) = (1 + \mathcal{O}(e^{-\alpha R}))$ . Due to the linearity of integration, constant factors within the integrand can be moved out of the integral, which yields

$$\begin{aligned} \mu(D(u)) &= \frac{\alpha}{\pi} e^{-\alpha R} (1 + \mathcal{O}(e^{-\alpha R})) \\ &\quad \cdot \int_{r(u)}^R \left( 2(e^{-r(u)/2} - e^{-r/2}) \pm \mathcal{O}(e^{-3/4 \cdot R}) \right) \cdot e^{\alpha r} \, dr \\ &= \frac{2\alpha}{\pi} e^{-r(u)/2} e^{-\alpha R} (1 + \mathcal{O}(e^{-\alpha R})) \int_{r(u)}^R e^{\alpha r} \, dr \\ &\quad - \frac{2\alpha}{\pi} e^{-\alpha R} (1 + \mathcal{O}(e^{-\alpha R})) \int_{r(u)}^R e^{(\alpha-1/2)r} \, dr \\ &\quad \pm \mathcal{O} \left( e^{-(3/4+\alpha)R} \int_{r(u)}^R e^{\alpha r} \, dr \right). \end{aligned}$$

The remaining integrals can be computed easily and we obtain

$$\begin{aligned} \mu(D(u)) &= \frac{2}{\pi} e^{-r(u)/2} (1 + \mathcal{O}(e^{-\alpha R})) (1 - e^{-\alpha(R-r(u))}) \\ &\quad - \frac{2\alpha}{(\alpha - 1/2)\pi} e^{-R/2} (1 + \mathcal{O}(e^{-\alpha R})) (1 - e^{-(\alpha-1/2)(R-r(u))}) \\ &\quad \pm \mathcal{O} \left( e^{-3/4 \cdot R} (1 - e^{-\alpha(R-r(u))}) \right). \end{aligned} \tag{3}$$

It remains to simplify the remaining error terms. To do this, we consider the three summands in the above expression separately, starting with the first. There, the error term can be expanded to obtain

$$\begin{aligned} (1 + \mathcal{O}(e^{-\alpha R})) (1 - e^{-\alpha(R-r(u))}) &= 1 + \mathcal{O}(e^{-\alpha R}) - e^{-\alpha(R-r(u))} - \mathcal{O}(e^{-\alpha R} \cdot e^{-\alpha(R-r(u))}) \\ &= 1 + e^{-\alpha R} \left( \mathcal{O}(1) - e^{\alpha r(u)} - \mathcal{O}(e^{-\alpha(R-r(u))}) \right) \end{aligned}$$

Now recall that  $R$  is defined as  $R = 2 \log(2n/(\pi\kappa) \cdot (\alpha/(\alpha - 1/2))^2(1 + o(1)))$ , which is equivalent to  $R = 2 \log(n) + C$  for some constant  $C \in \mathbb{R}$ , since  $\alpha$  and  $\kappa$  are assumed to be constants. Moreover, since  $r(u) \geq R/2$  holds by assumption, we have  $e^{\alpha r(u)} = \omega(1)$  and thus  $\mathcal{O}(1) - e^{\alpha r(u)} = -\mathcal{O}(e^{\alpha r(u)})$ . We obtain

$$(1 + \mathcal{O}(e^{-\alpha R}))(1 - e^{-\alpha(R-r(u))}) = 1 + e^{-\alpha R} \left( -\mathcal{O}(e^{\alpha r(u)}) - \mathcal{O}(e^{-\alpha(R-r(u))}) \right).$$

Again, since  $R = 2 \log(n) + C$  for a constant  $C$ , we have  $e^{-\alpha R} = o(1)$  and thus  $\mathcal{O}(e^{-\alpha(R-r(u))}) = \mathcal{O}(e^{\alpha r(u)})$ . Therefore, the error term further simplifies to  $(1 - \mathcal{O}(e^{-\alpha(R-r(u))}))$  and (3) becomes

$$\begin{aligned} \mu(D(u)) &= \frac{2}{\pi} e^{-r(u)/2} (1 - \mathcal{O}(e^{-\alpha(R-r(u))})) \\ &\quad - \frac{2\alpha}{(\alpha - 1/2)\pi} e^{-R/2} (1 + \mathcal{O}(e^{-\alpha R}))(1 - e^{-(\alpha-1/2)(R-r(u))}) \\ &\quad \pm \mathcal{O}\left(e^{-3/4 \cdot R} (1 - e^{-\alpha(R-r(u))})\right). \end{aligned}$$

Now consider the second summand. Since  $\alpha$  is constant, so is the first fraction. Moreover, as  $R = 2 \log(n) + C$  for a constant  $C$ , we have  $(1 + \mathcal{O}(e^{-\alpha R})) = (1 + o(1)) = \mathcal{O}(1)$ . And since  $r(u) \leq R$ , the exponent in the last factor is non-positive, from which we can conclude that this factor is also  $\mathcal{O}(1)$ . The second summand therefore simplifies to  $\mathcal{O}(e^{-R/2}) = \mathcal{O}(n^{-1})$ . Finally, the last summand can be reduced to  $\mathcal{O}(e^{-3/4 \cdot R}) = \mathcal{O}(n^{-3/2})$ , which yields

$$\mu(D(u)) = \frac{2}{\pi} e^{-r(u)/2} (1 - \mathcal{O}(e^{-\alpha(R-r(u))})) - \mathcal{O}(n^{-1}) \pm \mathcal{O}(n^{-3/2}).$$

Combining the last two summands then yields the claim. □

The following lemma shows that, with high probability, all vertices that are not too close to the boundary of the disk dominate at least one vertex.

**Lemma 3** *Let  $G$  be a hyperbolic random graph on  $n$  vertices, with power-law exponent  $2\alpha + 1$  and average degree  $\kappa$ . Then, there is a constant  $c > 2/(\kappa(1 - 1/(2\alpha))^2)$ , such that all vertices  $u$  with  $r(u) \leq \rho = R - 2 \log \log(n^c)$  are dominant, with high probability.*

*Proof* Vertex  $u$  is dominant if at least one vertex lies in  $D(u)$ . To show this for any  $u$  with  $r(u) \leq \rho$ , it suffices to show it for  $r(u) = \rho$ , since  $\mu(D(u))$  increases with

decreasing radius. To determine the probability that at least one vertex lies in  $D(u)$ , we use Lemma 2 and obtain

$$\begin{aligned} \mu(D(u)) &= \frac{2}{\pi} e^{-\rho/2} (1 - \mathcal{O}(e^{-\alpha(R-\rho)})) \pm \mathcal{O}(1/n) \\ &= \frac{2}{\pi} e^{-R/2 + \log \log(n^c)} (1 - \mathcal{O}(e^{-2\alpha \log \log(n^c)})) \pm \mathcal{O}(1/n). \end{aligned}$$

By substituting  $R = 2 \log(2n/(\pi\kappa) \cdot (\alpha/(\alpha - 1/2))^2(1 + o(1)))$ , we obtain

$$\mu(D(u)) = \frac{\kappa}{n} \left( \frac{\alpha - 1/2}{\alpha} \right)^2 \frac{1}{1 + o(1)} c \log(n) (1 - \mathcal{O}(\log(n)^{-2\alpha})) \pm \mathcal{O}(1/n).$$

Moreover, since  $1/(1 + x) = 1 - \Theta(x)$  for  $x \in \mathbb{R}$  with  $x = \pm o(1)$ , we can conclude that

$$\mu(D(u)) = c\kappa (1 - 1/(2\alpha))^2 \frac{\log(n)}{n} (1 - o(1)) \pm \mathcal{O}(1/n).$$

The probability of at least one vertex falling into  $D(u)$  is now given by

$$\begin{aligned} \Pr[\{v \in D(u)\} \neq \emptyset] &= 1 - (1 - \mu(D(u)))^n \\ &\geq 1 - e^{-n\mu(D(u))} \\ &= 1 - \Theta(n^{-c\kappa(1-1/(2\alpha))^2(1-o(1))}). \end{aligned}$$

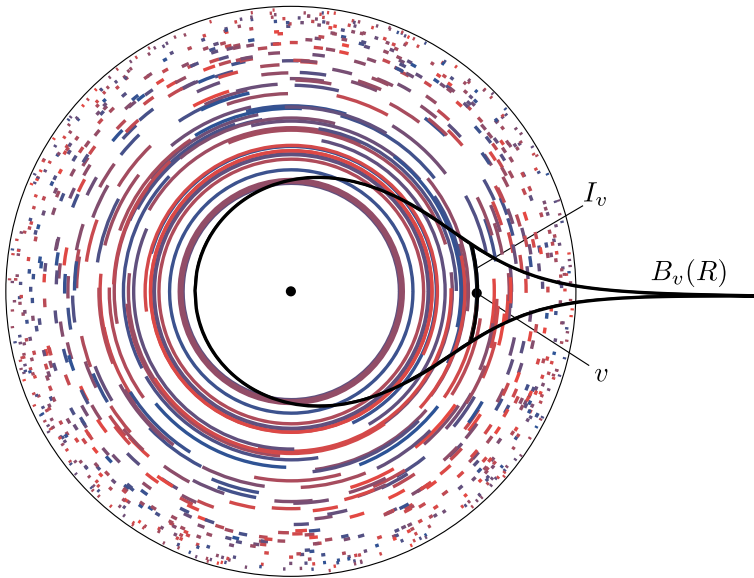
Consequently, for large enough  $n$  we can choose  $c > 2/(\kappa(1 - 1/(2\alpha))^2)$ , such that the probability of a vertex at radius  $\rho$  being dominant is at least  $1 - \Theta(n^{-2})$ , allowing us to apply the union bound. □

**Corollary 2** *Let  $G$  be a hyperbolic random graph on  $n$  vertices, with power-law exponent  $2\alpha + 1$  and average degree  $\kappa$ . Then, there exists a constant  $c > 2/(\kappa(1 - 1/(2\alpha))^2)$ , such that all vertices with radius at most  $\rho = R - 2 \log \log(n^c)$  are removed by the dominance rule, with high probability.*

By Corollary 2 the dominance rule removes all vertices of radius at most  $\rho$ . Consequently, all remaining vertices have radius at least  $\rho$ . We refer to this part of the disk as *outer band*. More precisely, the outer band is defined as  $B_O(R) \setminus B_O(\rho)$ . It remains to show that the pathwidth of the subgraph induced by the vertices in the outer band is small.

### 3.2 Pathwidth in the Outer Band

In the following, we use  $G|_{r(v) \geq r} = G[\{v \in V\} \mid r(v) \geq r]$  to denote the induced subgraph of  $G$  that contains all vertices with radius at least  $r$ . To show that the pathwidth of  $G|_{r(v) \geq \rho}$  (the induced subgraph in the outer band) is small, we first show that there is a circular arc supergraph  $\hat{G}|_{r(v) \geq \rho}$  of  $G|_{r(v) \geq \rho}$  with a small maximum clique. We use  $\hat{G}$  to denote a circular arc supergraph of a hyperbolic random graph  $G$ , which is obtained by assigning each vertex  $v$  an angular interval  $I_v$  on the circle, such that the intervals of two adjacent vertices intersect. More precisely, for a vertex  $v$ , we set  $I_v = [\varphi(v) - \theta(r(v), r(v)), \varphi(v) + \theta(r(v), r(v))]$ . Intuitively, this



**Fig. 4** The angular intervals representing the circular arc supergraph  $\hat{G}$  of a hyperbolic random graph  $G$ . The arc  $I_v$  of a vertex  $v$  extends to the boundary of its neighborhood disk  $B_v(R)$  at the radius of  $v$

means that the interval of a vertex contains a superset of all its neighbors that have a larger radius, as can be seen in Fig. 4. The following lemma shows that  $\hat{G}$  is actually a supergraph of  $G$ .

**Lemma 4** *Let  $G = (V, E)$  be a hyperbolic random graph. Then  $\hat{G}$  is a supergraph of  $G$ .*

*Proof* Let  $\{u, v\} \in E$  be any edge in  $G$ . To show that  $\hat{G}$  is a supergraph of  $G$  we need to show that  $u$  and  $v$  are also adjacent in  $\hat{G}$ , i.e.,  $I_u \cap I_v \neq \emptyset$ . Without loss of generality assume  $r(u) \leq r(v)$ . Since  $u$  and  $v$  are adjacent in  $G$ , the hyperbolic distance between them is at most  $R$ . It follows, that their angular distance  $\Delta_\varphi(u, v)$  is bounded by  $\theta(r(u), r(v))$ . Since  $\theta(r(u), r(v)) \leq \theta(r(u), r(u))$  for  $r(u) \leq r(v)$ , we have  $\Delta_\varphi(u, v) \leq \theta(r(u), r(u))$ . As  $I_u$  extends by  $\theta(r(u), r(u))$  from  $\varphi(u)$  in both directions, it follows that  $\varphi(v) \in I_u$ .  $\square$

Note that  $\hat{G}$  is still a supergraph of  $G$ , after removing a vertex from both  $G$  and  $\hat{G}$ . Consequently,  $\hat{G}|_{r(v) \geq \rho}$  is a supergraph of  $G|_{r(v) \geq \rho}$ . It remains to show that  $\hat{G}|_{r(v) \geq \rho}$  has a small maximum clique number, which is given by the maximum number of arcs that intersect at any angle. To this end, we first compute this number at a given angle, which we set to 0 without loss of generality. Let  $A_r$  denote the area of the disk containing all vertices  $v$  with radius  $r(v) \geq r$  whose interval  $I_v$  intersects 0, as illustrated in Fig. 5. The following lemma describes the probability for a given vertex to lie in  $A_r$ .

**Lemma 5** *Let  $G$  be a hyperbolic random graph and let  $r \geq R/2$ . The probability for a given vertex to lie in  $A_r$  is bounded by*

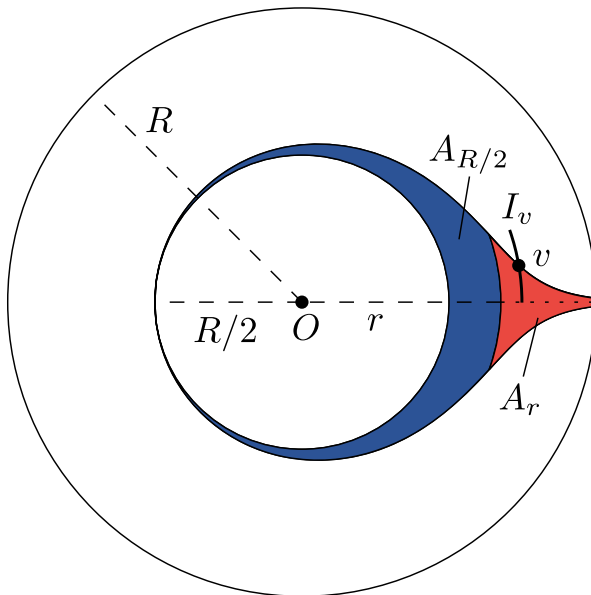
$$\mu(A_r) \leq \frac{2\alpha}{(1-\alpha)\pi} e^{-(\alpha-1/2)R-(1-\alpha)r} \cdot \left(1 + \mathcal{O}(e^{-\alpha R} + e^{-(2r-R)}) - \mathcal{O}(e^{-(1-\alpha)(R-r)})\right).$$

*Proof* We obtain the measure of  $A_r$  by integrating the probability density function over  $A_r$ . Due to the definition of  $I_v$  we can conclude that  $A_r$  includes all vertices  $v$  with radius  $r(v) \geq r$  whose angular distance to 0 is at most  $\theta(r(v), r(v))$ , defined in (2). We obtain,

$$\begin{aligned} \mu(A_r) &= \int_r^R 2 \int_0^{\theta(x,x)} f(x, \varphi) \, d\varphi \, dx \\ &= 2 \int_r^R \left( 2e^{(R-2x)/2} (1 \pm \Theta(e^{R-2x})) \cdot \frac{\alpha}{2\pi} e^{-\alpha(R-x)} (1 + \Theta(e^{-\alpha R} - e^{-2\alpha x})) \right) dx. \end{aligned}$$

As before, we can conclude that  $(1 + \Theta(e^{-\alpha R} - e^{-2\alpha r})) = (1 + \mathcal{O}(e^{-\alpha R}))$ , since  $r \geq R/2$ . By moving constant factors out of the integral, the expression can be simplified to

$$\mu(A_r) \leq \frac{2\alpha}{\pi} e^{-(\alpha-1/2)R} (1 + \mathcal{O}(e^{-\alpha R})) \int_r^R e^{-(1-\alpha)x} (1 + \Theta(e^{R-2x})) \, dx.$$



**Fig. 5** The area that contains the vertices whose arcs intersect angle 0. Area  $A_r$  (red) contains all such vertices with radius at least  $r$ . Vertex  $v$  lies on the boundary of  $A_r$  and its interval  $I_v$  extends to 0

We split the sum in the integral and deal with the resulting integrals separately.

$$\begin{aligned} \mu(A_r) &\leq \frac{2\alpha}{\pi} e^{-(\alpha-1/2)R} (1 + \mathcal{O}(e^{-\alpha R})) \\ &\quad \cdot \left( \int_r^R e^{-(1-\alpha)x} dx + \Theta \left( \int_r^R e^{-(1-\alpha)x+R-2x} dx \right) \right) \\ &= \frac{2\alpha}{\pi} e^{-(\alpha-1/2)R} (1 + \mathcal{O}(e^{-\alpha R})) \\ &\quad \cdot \left( \frac{1}{1-\alpha} e^{-(1-\alpha)r} (1 - e^{-(1-\alpha)(R-r)}) \right. \\ &\quad \left. + \Theta \left( e^R e^{-(3-\alpha)r} (1 - e^{-(3-\alpha)(R-r)}) \right) \right). \end{aligned}$$

By placing  $1/(1-\alpha) \cdot e^{-(1-\alpha)r}$  outside of the parentheses we obtain

$$\begin{aligned} \mu(A_r) &\leq \frac{2\alpha}{(1-\alpha)\pi} e^{-(\alpha-1/2)R-(1-\alpha)r} (1 + \mathcal{O}(e^{-\alpha R})) \\ &\quad \cdot \left( (1 - e^{-(1-\alpha)(R-r)}) + \Theta(e^{R-2r} (1 - e^{-(3-\alpha)(R-r)})) \right). \end{aligned}$$

Simplifying the remaining error terms then yields the claim. □

We can now bound the maximum clique number in  $\hat{G}|_{r(v) \geq \rho}$  and with that its interval width  $\text{iw}(\hat{G}|_{r(v) \geq \rho})$ .

**Theorem 3** *Let  $G$  be a hyperbolic random graph on  $n$  vertices and let  $r \geq R/2$ . Then there exists a constant  $c$  such that, with high probability, it holds that  $\text{iw}(\hat{G}|_{r(v) \geq r}) = \mathcal{O}(\log(n))$ , if  $r \geq R - 1/(1-\alpha) \cdot \log \log(n^c)$ , and otherwise*

$$\begin{aligned} \text{iw}(\hat{G}|_{r(v) \geq r}) &\leq \frac{5\alpha}{(1-\alpha)\pi} n e^{-(\alpha-1/2)R-(1-\alpha)r} \\ &\quad \cdot (1 + \mathcal{O}(e^{-\alpha R} + e^{-(2r-R)}) - \mathcal{O}(e^{-(1-\alpha)(R-r)})). \end{aligned}$$

*Proof* We start by determining the expected number of arcs that intersect at a given angle, which can be done by computing the expected number of vertices in  $A_r$ , using Lemma 5:

$$\begin{aligned} \mathbb{E}[|\{v \in A_r\}|] &\leq \frac{2\alpha}{(1-\alpha)\pi} n e^{-(\alpha-1/2)R-(1-\alpha)r} \\ &\quad \cdot (1 + \mathcal{O}(e^{-\alpha R} + e^{-(2r-R)}) - \mathcal{O}(e^{-(1-\alpha)(R-r)})) \\ &=: g(r). \end{aligned}$$

It remains to show that this bound holds with high probability at every angle. To this end, we apply a Chernoff bound (Corollary 1) to conclude that for any  $\varepsilon \in (0, 1)$  it holds that

$$\Pr[|\{v \in A_r\}| > (1 + \varepsilon)g(r)] \leq e^{-\varepsilon^2/3 \cdot g(r)}.$$

In order to see that this probability is sufficiently small, we first take a closer look at  $g(r')$  with  $r' = R - 1/(1 - \alpha) \cdot \log \log(n^c)$  and afterwards argue about the different values that  $r$  can take relative to  $r'$ .

$$\begin{aligned} g(r') &= \frac{2\alpha}{(1 - \alpha)\pi} n e^{-(\alpha-1/2)R - (1-\alpha)(R-1/(1-\alpha) \cdot \log \log(n^c))} \\ &\quad \cdot (1 + \mathcal{O}(e^{-\alpha R} + e^{-(2(R-1/(1-\alpha) \cdot \log \log(n^c)) - R)}) \\ &\quad \quad - \mathcal{O}(e^{-(1-\alpha)(R - (R-1/(1-\alpha) \log \log(n^c))})) \\ &= \frac{2\alpha}{(1 - \alpha)\pi} n e^{-R/2 + \log \log(n^c)} \\ &\quad \cdot (1 + \Theta(e^{-\alpha R} + e^{-(R-2/(1-\alpha) \cdot \log \log(n^c))}) - \mathcal{O}(e^{-\log \log(n^c)})) \end{aligned}$$

Substituting  $R = 2 \log(2n/(\pi\kappa) \cdot (\alpha/(\alpha - 1/2))^2(1 + o(1)))$  we obtain

$$g(r') = c\kappa \frac{(\alpha - 1/2)^2}{(1 - \alpha)\alpha} \log(n)(1 \pm o(1)).$$

Now consider the case where  $r < r'$ . Then,  $g(r) > g(r')$  and applying Corollary 1 with  $\varepsilon = 1/4$  yields

$$\Pr[|\{v \in A_r\}| > 5/4 \cdot g(r)] \leq e^{-\frac{\varepsilon^2}{3} g(r)} \leq e^{-\frac{1}{48} g(r')} \leq n^{-c\kappa \frac{(\alpha-1/2)^2}{48(1-\alpha)\alpha} (1 \pm o(1))}.$$

For the case, where  $r \geq r'$ , note that  $\mathbb{E}[|\{v \in A_r\}|]$  decreases with increasing  $r$ . Therefore,  $g(r') \in \mathcal{O}(\log(n))$  is a pessimistic but valid upper bound on  $g(r)$  and we obtain the same bound on  $\Pr[|\{v \in A_r\}| > 5/4 \cdot g(r')]$ .

In both cases, we can choose  $c$  such that  $|\{v \in A_r\}| \leq 5/4 \cdot g(r)$  holds with probability  $1 - \mathcal{O}(n^{-c'})$  for any  $c'$  at a given angle. In order to see that it holds at every angle, note that it suffices to show that it holds at all arc endings as the number of intersecting arcs does not change in between arc endings. Since there are exactly  $2n$  arc endings, we can apply the union bound and obtain that the bound holds with probability  $1 - \mathcal{O}(n^{-c'+1})$  for any  $c'$  at every angle. Since  $g(r)$  is an upper bound on the maximum clique size of  $\hat{G}|_{r(v) \geq r}$ , the interval width of  $\hat{G}|_{r(v) \geq r}$  is at most twice as large, as argued in Section 2. □

Since the interval width of a circular arc supergraph of  $G$  is an upper bound on the pathwidth of  $G$  [13, Theorem 7.14] and since  $\rho \geq R - 1/(1 - \alpha) \cdot \log \log(n^c)$  for  $\alpha \in (1/2, 1)$ , we immediately obtain the following corollary.

**Corollary 3** *Let  $G$  be a hyperbolic random graph on  $n$  vertices and let  $G|_{r(v) \geq \rho}$  be the subgraph obtained by removing all vertices with radius at most  $\rho = R - 2 \log \log(n^c)$ . Then, with high probability it holds that*

$$pw(G|_{r(v) \geq \rho}) = \mathcal{O}(\log(n)).$$

We are now ready to prove our main theorem, which we restate for the sake of readability.



**Theorem 4** *Let  $G$  be a hyperbolic random graph on  $n$  vertices. Then the VERTEXCOVER problem in  $G$  can be solved in  $\text{poly}(n)$  time, with high probability.*

*Proof* Consider the following algorithm that finds a minimum vertex cover of  $G$ . We start with an empty vertex cover  $S$ . Initially, all dominant vertices are added to  $S$ , which is correct due to the dominance rule. By Lemma 3, this includes all vertices of radius at most  $\rho = R - 2 \log \log(n^c)$ , for some constant  $c$ , with high probability. Obviously, finding all vertices that are dominant can be done in  $\text{poly}(n)$  time. It remains to determine a vertex cover of  $G|_{r(v) \geq \rho}$ . By Corollary 3, the pathwidth of  $G|_{r(v) \geq \rho}$  is  $\mathcal{O}(\log(n))$ , with high probability. Since the pathwidth is an upper bound on the treewidth, we can find a tree decomposition of  $G|_{r(v) \geq \rho}$  and solve the VERTEXCOVER problem in  $G|_{r(v) \geq \rho}$  in  $\text{poly}(n)$  time [13, Theorems 7.18 and 7.9]. □

Moreover, linking the radius of a vertex in Theorem 3 with its expected degree leads to the following corollary, which is interesting in its own right. It links the pathwidth to the degree  $d$  in the graph  $G|_{\text{deg}(v) \leq d} = G[\{v \in V \mid \text{deg}(v) \leq d\}]$ , i.e., the subgraph of  $G$  induced by vertices of degree at most  $d$ .

**Corollary 4** *Let  $G$  be a hyperbolic random graph and let  $d \leq \sqrt{n}$ . Then, with high probability,  $\text{pw}(G|_{\text{deg}(v) \leq d}) = \mathcal{O}(d^{2-2\alpha} + \log(n))$ .*

*Proof* Consider the radius  $r = R - 2 \log(\xi d)$  for some constant  $\xi > 0$ , and the graph  $G|_{r(v) \geq r}$  that is obtained by removing all vertices of radius at most  $r$ . In the following, we show that  $G|_{r(v) \geq r}$  is a supergraph of  $G|_{\text{deg}(v) \leq d}$  for large enough  $\xi$ . Afterwards, we bound the pathwidth of  $G|_{r(v) \geq r}$ .

The expected degree of a vertex with radius  $r$  is given by

$$\mathbb{E}[\text{deg}(v) \mid r(v) = r] = \frac{2\alpha}{(\alpha - 1/2)\pi} n e^{-r/2} (1 \pm \mathcal{O}(e^{-(\alpha-1/2)r})). \quad ([18, \text{Theorem 3.2}])$$

By substituting  $r = R - 2 \log(\xi d)$  together with the expression for  $R$ , which is given by  $R = 2 \log(2n/(\pi\kappa) \cdot (\alpha/(\alpha - 1/2))^2(1 + o(1)))$ , we obtain

$$\begin{aligned} \mathbb{E}[\text{deg}(v) \mid r(v) = r] &= \frac{2\alpha}{(\alpha - 1/2)\pi} n e^{-R/2 + \log(\xi d)} \\ &\quad \cdot (1 \pm \mathcal{O}(e^{-(\alpha-1/2)(R-2\log(\xi d))})) \\ &= \frac{2\alpha\kappa}{2(\alpha - 1/2)} \left( \frac{\alpha - 1/2}{\alpha} \right)^2 \frac{1}{1 + o(1)} \xi \cdot d \\ &\quad \cdot (1 \pm \mathcal{O}((d/n)^{(2\alpha-1)})) \\ &= \xi\kappa(1 - 1/(2\alpha)) \cdot d(1 \pm o(1)). \end{aligned}$$

Note that for large enough  $n$  we can choose  $\xi$  sufficiently large, such that

$$\Pr[\text{deg}(v) \leq d \mid r(v) = r] \leq \Pr[\text{deg}(v) < (1 - \varepsilon)\mathbb{E}[\text{deg}(v) \mid r(v) = r]],$$

for any  $\varepsilon \in (0, 1)$ . This allows us to apply the second inequality in the Chernoff bound in Theorem 1 to conclude that

$$\Pr[\text{deg}(v) \leq d \mid r(v) = r] \leq \exp\left(-\varepsilon^2/2 \cdot \xi k(1 - 1/(2\alpha)) \cdot d(1 \pm o(1))\right).$$

First assume that  $d \geq \log(n)^{1/(2-2\alpha)}$ . We handle the other case later. Note that  $1/(2-2\alpha) > 1$  for  $\alpha \in (1/2, 1)$  and, thus,  $d \geq \log(n)$ . Therefore, we can choose  $n$  and  $\xi$  sufficiently large, such that

$$\Pr[\text{deg}(v) \leq d \mid r(v) = r] \leq n^{-\frac{\varepsilon^2}{2} \xi k(1-1/(2\alpha))(1\pm o(1))} \leq n^{-2}.$$

Since smaller radius implies larger expected degree, we can derive the same bound for a given vertex of radius *at most*  $r$ . By applying the union bound we obtain that, with high probability, no vertex with radius at most  $r$  has degree less than or equal to  $d$ . Conversely, all vertices with degree at most  $d$  have radius at least  $r$ . Consequently,  $\hat{G}|_{r(v) \leq r}$  is a supergraph of  $\hat{G}|_{\text{deg}(v) \leq d}$ .

To prove the claim, it remains to bound the pathwidth of  $G|_{r(v) \geq r}$ . If  $r > R - 1/(1 - \alpha) \cdot \log \log(n^c)$ , we can apply the first part of Theorem 3 to obtain  $\text{iw}(\hat{G}|_{r(v) \geq r}) = \mathcal{O}(\log(n))$ . Otherwise, we use part two to conclude that the interval width of  $G|_{r(v) \geq r}$  is at most

$$\begin{aligned} \text{iw}(\hat{G}|_{r(v) \geq r}) &\leq \frac{5\alpha}{(1 - \alpha)\pi} n e^{-(\alpha-1/2)R - (1-\alpha)r} \\ &\quad \cdot (1 + \mathcal{O}(e^{-\alpha R} + e^{-(2r-R)}) - \mathcal{O}(e^{-(1-\alpha)(R-r)})) \\ &= \frac{5\kappa\alpha\xi^{2-2\alpha}}{2(1 - \alpha)} \left(\frac{\alpha - 1/2}{\alpha}\right)^2 \frac{1}{(1 + o(1))} \\ &\quad \cdot (1 + \mathcal{O}(n^{-2\alpha} + (d^2/n)^2) - \mathcal{O}(d^{-(2-2\alpha)})) \\ &= \frac{5\kappa(\alpha - 1/2)^2 \xi^{2-2\alpha}}{2(1 - \alpha)\alpha} d^{2-2\alpha} (1 \pm \mathcal{O}(1)) \\ &= \mathcal{O}(d^{2-2\alpha}). \end{aligned}$$

As argued in Section 2 the interval width is an upper bound on the pathwidth.

For the case where  $d < \log(n)^{1/(2-2\alpha)}$  (which we excluded above), consider  $G|_{\text{deg}(v) \leq d'}$  for  $d' = \log(n)^{1/(2-2\alpha)} > d$ . As we already proved the corollary for  $d'$ , we obtain  $\text{pw}(G|_{\text{deg}(v) \leq d'}) = \mathcal{O}(d'^{2-2\alpha} + \log(n)) = \mathcal{O}(\log(n))$ . As  $G|_{\text{deg}(v) \leq d}$  is a subgraph of  $G|_{\text{deg}(v) \leq d'}$ , the same bound holds for  $G|_{\text{deg}(v) \leq d}$ .  $\square$

### 4 Empirical Evaluation

Our results show that a heterogeneous degree distribution as well as high clustering make the dominance rule very effective. This matches the behavior for real-world net-

works, which typically exhibit these two properties. However, our analysis actually makes more specific predictions: (I) vertices with sufficiently high degree usually have at least one neighbor they dominate and can thus safely be included in the vertex cover; and (II) the graph remaining after deleting the high-degree vertices has simple structure, i.e., small pathwidth.

To see whether this matches the real world, we ran experiments on 59 networks from several network datasets [4, 5, 21, 22, 24]. Although the focus of this paper is on the theoretical analysis on hyperbolic random graphs, we briefly report on our experimental results; see Table 1 in Appendix. Out of the 59 instances, we can solve VERTEXCOVER for 47 networks in reasonable time. We refer to these as *easy*, while the remaining 12 are called *hard*. Note that our theoretical analysis aims at explaining why the easy instances are easy.

Recall from Lemma 3 that all vertices with radius at most  $R - 2 \log \log(n^c)$ , with  $c > 2/(\kappa(1 - 1/(2\alpha))^2)$ , probably dominate. This corresponds to an expected degree of  $2\alpha/(\alpha - 1/2) \cdot \log(n)$ . Figure 6 shows the percentage of dominant vertices among the ones above this degree, for the considered real-world networks. For more than 66% of the 59 networks, more than 75% of these vertices were in fact dominant (red and blue). For more than 40% of the networks, more than 95% were dominant (blue). Restricted to the 47 easy instances, these increase to 82% and 51% of networks, respectively.

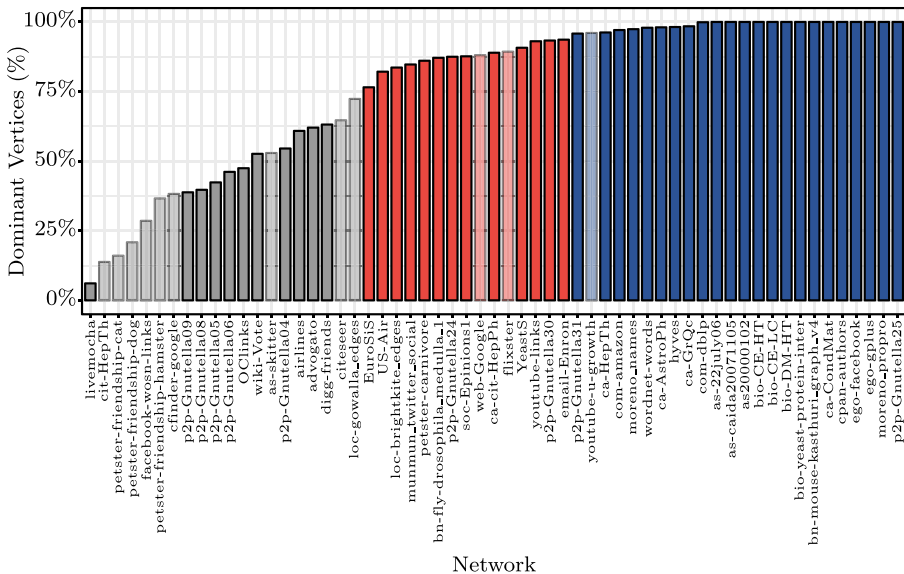
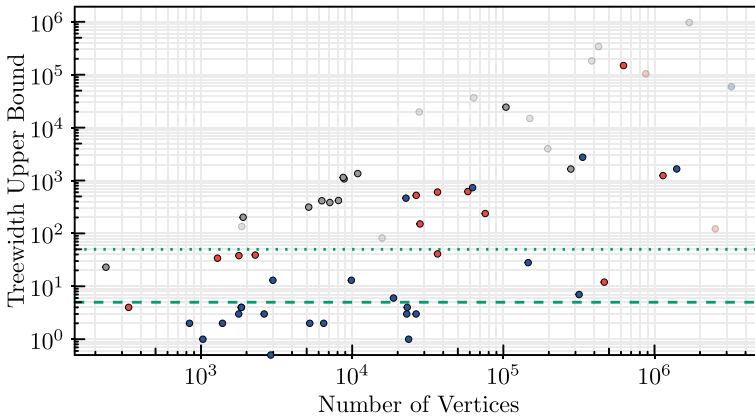


Fig. 6 Percentage of dominant vertices among ones with degree above  $2\alpha/(\alpha - 1/2) \log(n)$ . Red and blue bars denote networks where this value is above 75%. Blue bars denote networks where it is above 95%. Transparent bars denote hard instances



**Fig. 7** Upper bounds on the treewidth of the considered graphs, after removing initially dominant vertices. Dashed and dotted green lines denote a bound of 5 and 50, respectively. Colors represent the percentage of initially dominant high-degree vertices, analogous to Fig. 6. Transparent dots represent hard instances

Experiments concerning the pathwidth of the resulting graph are much more difficult, due to the lack of efficient tools. Therefore, we used the tool by Tamaki et al. [25] to heuristically compute upper bounds on the treewidth instead. As in our analysis, we only removed vertices that dominate in the original graph instead of applying the reduction rule exhaustively. On the resulting subgraphs, the treewidth heuristic ran with a 15min timeout. The resulting treewidth is at most 50 for 44% of the networks and at most 5 for 25%, see Fig. 7. Restricted to easy instances, the values increase to 55% and 32%, respectively. Note how on most graphs where almost all high-degree vertices are dominant (blue), we obtained the smallest treewidths. This indicates, that on networks where our first prediction was fulfilled, so was the second one.

While hyperbolic random graphs are clearly an idealized representation of real-world graphs, these experiments indicate that the predictions derived from the model match the real world, at least for a significant fraction of networks.

## Appendix: Experimental Data

Table 1 (continuing on the next page) shows the raw data of our experiments for which we reported aggregate values in the discussion in Section 4. The percentage of dominant vertices among those with high degree ( $> 2\alpha/(\alpha - 1/2) \cdot \log n$ ) is rounded to whole percentages. Treewidth  $-1$  indicates that the remaining graph after removing all dominant vertices contained no edge.

**Table 1** The raw data of our experiments

Network	Vertices	Easy	Dominant	Treewidth
advogato	5158	✓	62%	314
airlines	235	✓	61%	23
as-22july06	22963	✓	100%	3
as-caida20071105	26478	✓	100%	3
as-skitter	1696418	×	53%	969794
as20000102	6477	✓	100%	2
bio-CE-HT	2617	✓	100%	3
bio-CE-LC	1387	✓	100%	2
bio-DM-HT	2989	✓	100%	13
bio-yeast-protein-inter	1846	✓	100%	4
bn-fly-drosophila_medulla_1	1781	✓	87%	38
bn-mouse-kasthuri_graph_v4	1029	✓	100%	1
ca-AstroPh	18771	✓	98%	6
ca-cit-HepPh	28096	✓	89%	151
ca-CondMat	23133	✓	100%	4
ca-GrQc	5241	✓	98%	2
ca-HepTh	9875	✓	96%	13
cfinder-google	15766	×	38%	82
cit-HepTh	27769	×	14%	19737
citeseer	384057	×	65%	182372
com-amazon	334866	✓	97%	2756
com-dblp	317083	✓	100%	7
cpan-authors	839	✓	100%	2
digg-friends	279632	✓	63%	1649
ego-facebook	2891	✓	100%	-1
ego-gplus	23631	✓	100%	1
email-Enron	36692	✓	94%	41
EuroSiS	1285	✓	77%	34
facebook-wosn-links	63734	×	29%	36694
flixster	2523388	×	89%	122
hyves	1402675	✓	98%	1653
livemocha	104105	✓	6%	24380
loc-brightkite_edges	58231	✓	84%	619
loc-gowalla_edges	196594	×	72%	3991
moreno_names	1776	✓	97%	3
moreno_propro	1849	✓	100%	4
munmun_twitter_social	465020	✓	85%	12
OCLinks	1899	✓	47%	202
p2p-Gnutella04	10876	✓	55%	1352
p2p-Gnutella05	8846	✓	42%	1075

**Table 1** (continued)

Network	Vertices	Easy	Dominant	Treewidth
p2p-Gnutella06	8717	✓	46%	1142
p2p-Gnutella08	6301	✓	40%	414
p2p-Gnutella09	8114	✓	39%	419
p2p-Gnutella24	26518	✓	88%	525
p2p-Gnutella25	22687	✓	100%	464
p2p-Gnutella30	36682	✓	93%	604
p2p-Gnutella31	62586	✓	96%	732
petster-carnivore	623750	✓	86%	149312
petster-friendship-cat	149686	×	16%	14929
petster-friendship-dog	426818	×	21%	340634
petster-friendship-hamster	1860	×	37%	135
soc-Epinions1	75882	✓	88%	238
US-Air	332	✓	82%	4
web-Google	875716	×	88%	103939
wiki-Vote	7115	✓	53%	384
wordnet-words	146008	✓	98%	28
YeastS	2284	✓	91%	39
youtube-links	1138497	✓	93%	1239
youtube-u-growth	3223588	×	96%	59358

The columns are: **(Network)** the network's name; **(Vertices)** the number of vertices in the network; **(Easy)** whether or not we could compute an optimal solution; **(Dominant)** the percentage of dominant vertices among high-degree vertices; **(Treewidth)** an upper bound for the treewidth of the remaining graph after deleting dominant vertices

**Acknowledgements** This research was partially funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) – project number 390859508.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Akiba, T., Iwata, Y.: Branch-and-Reduce Exponential/FPT Algorithms in Practice: A Case Study of Vertex Cover. *Theor. Comput. Sci.* **609**, 211–225 (2016). <https://doi.org/10.1016/j.tcs.2015.09.023>

2. Albert, R.: Scale-Free networks in cell biology. *J. Cell Sci.* **118**(21), 4947–4957 (2005). <https://doi.org/10.1242/jcs.02714>
3. Albert, R., Barabási, A.L.: Statistical Mechanics of Complex Networks. *Rev. Mod. Phys.* **74**, 47–97 (2002). <https://doi.org/10.1103/RevModPhys.74.47>
4. Arenas, A., Barabási, A.L., Batagelj, V., Mrvar, A., Newman, M., Opsahl, T.: Gephi Datasets. <https://github.com/gephi/gephi/wiki/Datasets>
5. Batagelj, V., Mrvar, A.: Pajek datasets. <http://vlado.fmf.uni-lj.si/pub/networks/data/> (2006)
6. Bläsius, T., Fischbeck, P., Friedrich, T., Katzmann, M.: Solving Vertex Cover in Polynomial Time on Hyperbolic Random Graphs. In: 37Th International Symposium on Theoretical Aspects of Computer Science, STACS, pp. 25:1–25:14. Montpellier, France (2020). <https://doi.org/10.4230/LIPIcs.STACS.2020.25>
7. Bläsius, T., Freiberger, C., Friedrich, T., Katzmann, M., Montenegro-Retana, F., Thieffry, M.: Efficient Shortest Paths in Scale-Free Networks with Underlying Hyperbolic Geometry. In: 45Th International Colloquium on Automata, Languages, and Programming (ICALP), pp. 20:1–20:14 (2018). <https://doi.org/10.4230/LIPIcs.ICALP.2018.20>
8. Bläsius, T., Friedrich, T., Katzmann, M.: Efficiently Approximating Vertex Cover on Scale-Free Networks with Underlying Hyperbolic Geometry. To appear in the proceedings of the 29th Annual European Symposium on Algorithms (ESA) (2021)
9. Bläsius, T., Friedrich, T., Krohmer, A.: Hyperbolic Random Graphs: Separators and Treewidth. In: 24Th Annual European Symposium on Algorithms (ESA), pp. 15:1–15:16 (2016). <https://doi.org/10.4230/LIPIcs.ESA.2016.15>
10. Boguná, M., Papadopoulos, F., Krioukov, D.: Sustaining the Internet with Hyperbolic Mapping. *Nat. Commun.* **1**, 62 (2010). <https://doi.org/10.1038/ncomms1063>
11. Cai, L., Juedes, D.: On the Existence of Subexponential Parameterized Algorithms. *J. Comput. Syst. Sci.* **67**, 789–807 (2003). [https://doi.org/10.1016/S0022-0000\(03\)00074-6](https://doi.org/10.1016/S0022-0000(03)00074-6)
12. Chen, J., Kanj, I.A., Xia, G.: Improved Upper Bounds for Vertex Cover. *Theor. Comput. Sci.* **411**(40), 3736–3756 (2010). <https://doi.org/10.1016/j.tcs.2010.06.026>
13. Cygan, M., Fomin, F.V., Kowalik, Ł., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: *Parameterized Algorithms*. Springer (2015)
14. Dorogovtsev, S.: *Lectures on Complex Networks*. Oxford University Press, Inc. <https://doi.org/10.1093/acprof:oso/9780199548927.001.0001> (2010)
15. Dubhashi, D.P., Panconesi, A.: *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press (2012)
16. Fomin, F.V., Grandoni, F., Kratsch, D.: A Measure & Conquer Approach for the Analysis of Exact Algorithms. *J. ACM* **56**(5), 25:1–25:32 (2009). <https://doi.org/10.1145/1552285.1552286>
17. Friedrich, T., Krohmer, A.: On the Diameter of Hyperbolic Random Graphs. *SIAM J. Discret. Math.* **32**(2), 1314–1334 (2018). <https://doi.org/10.1137/17M1123961>
18. Gugelmann, L., Panagiotou, K., Peter, U.: Random Hyperbolic Graphs: Degree Sequence and Clustering. In: *Automata, Languages, and Programming*, pp. 573 – 585. Springer Berlin Heidelberg. [https://doi.org/10.1007/978-3-642-31585-5\\_51](https://doi.org/10.1007/978-3-642-31585-5_51) (2012)
19. Kiwi, M.A., Mitsche, D.: A Bound for the Diameter of Random Hyperbolic Graphs. In: *Proceedings of the Twelfth Workshop on Analytic Algorithmics and Combinatorics, ANALCO*, pp. 26–39. SIAM (2015). <https://doi.org/10.1137/1.9781611973761.3>
20. Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A., Boguná, M.: Hyperbolic Geometry of Complex Networks. *Phys. Rev. E* **82**, 036106 (2010). <https://doi.org/10.1103/PhysRevE.82.036106>
21. Kunegis, J.: KONECT: The Koblenz Network Collection. In: *International Conference on World Wide Web (WWW)*, Pp. 1343 – 1350. <https://doi.org/10.1145/2487788.2488173> (2013)
22. Leskovec, J., Krevl, A.: SNAP Datasets: Stanford Large Network Dataset Collection. <http://snap.stanford.edu/data> (2014)
23. Ramsay, A., Richtmyer, R.D.: *Introduction to Hyperbolic Geometry*. Springer. <https://doi.org/10.1007/978-1-4757-5585-5> (1995)
24. Rossi, R.A., Ahmed, N.K.: The Network Data Repository with Interactive Graph Analytics and Visualization. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. <http://networkrepository.com> (2015)

25. Tamaki, H., Ohtsuka, H., Sato, T., Makii, K.: TCS-meiji PACE2017-tracka [github.com/TCS-meiji/PACE2017-tracka](https://github.com/TCS-meiji/PACE2017-tracka) (2017)
26. Xiao, M., Nagamochi, H.: Exact Algorithms for Maximum Independent Set. *Inf. Comput.* **255**, 126–146 (2017). <https://doi.org/10.1016/j.ic.2017.06.001>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

Thomas Bläsius<sup>1</sup> · Philipp Fischbeck<sup>2</sup> · Tobias Friedrich<sup>2</sup>  · Maximilian Katzmann<sup>3</sup> 

Thomas Bläsius  
thomas.blaesius@kit.edu

Philipp Fischbeck  
philipp.fischbeck@hpi.de

Tobias Friedrich  
tobias.friedrich@hpi.de

<sup>1</sup> Karlsruhe Institute of Technology, Karlsruhe, Germany

<sup>2</sup> Hasso Plattner Institute, University of Potsdam, Potsdam, Germany

<sup>3</sup> Algorithm Engineering, Hasso Plattner Institute, University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Brandenburg, Germany