Greed is Good for Deterministic Scale-Free Networks

Ankit Chauhan · Tobias Friedrich ·
Ralf Rothenberger

Abstract Large real-world networks typically follow a power-law degree distribution. To study such networks, numerous random graph models have been proposed. However, real-world networks are not drawn at random. Therefore, Brach, Cygan, Lacki, and Sankowski [SODA 2016] introduced two natural deterministic conditions: (1) a power-law upper bound on the degree distribution (PLB-U) and (2) power-law neighborhoods, that is, the degree distribution of neighbors of each vertex is also upper bounded by a power law (PLB-N). They showed that many real-world networks satisfy both properties and exploit them to design faster algorithms for a number of classical graph problems.

We complement their work by showing that some well-studied random graph models exhibit both of the mentioned PLB properties. PLB-U and PLB-N hold with high probability for Chung-Lu Random Graphs and Geometric Inhomogeneous Random Graphs and almost surely for Hyperbolic Random Graphs. As a consequence, all results of Brach et al. also hold with high probability or almost surely for those random graph classes.

In the second part we study three classical NP-hard optimization problems on PLB networks. It is known that on general graphs with maximum degree Δ, a greedy algorithm, which chooses nodes in the order of their degree, only achieves a $\Omega(\ln \Delta)$-approximation for MINIMUM VERTEX COVER and MINIMUM DOMINATING SET, and a $\Omega(\Delta)$-approximation for MAXIMUM INDEPENDENT SET. We prove that the PLB-U property with $\beta > 2$ suffices for the greedy approach to achieve a constant-factor approximation for all three problems. We also show that these problems are APX-hard even if PLB-U, PLB-N, and an additional power-law lower bound on the degree distribution hold. Hence, a PTAS cannot be expected unless P=NP. Furthermore, we prove that all three problems are in MAX SNP if the PLB-U property holds.

Ankit Chauhan · Tobias Friedrich · Ralf Rothenberger
Hasso Plattner Institute, Potsdam, Germany
E-mail: firstname.lastname@hpi.de

1 Introduction

A wide range of real-world networks exhibit a degree distribution that resembles a power-law [4, 39]. This means that the number of vertices with degree k is proportional to $k^{-\beta}$, where $\beta > 1$ is the power-law exponent, a constant intrinsic to the network. This applies to Internet topologies [24], the Web [7, 35], social networks [1], power grids [43], and literally hundreds of other domains [40]. Networks with a power-law degree distribution are also called scale-free networks and have been widely studied.

To capture the degree distribution and other properties of scale-free networks, a multitude of random graph models have been proposed. These models include Preferential Attachment [7], the Configuration Model [2], Chung-Lu Random Graphs [18] and Hyperbolic Random Graphs [34]. Despite the multitude of random models, none of the models truly has the same set of properties as real world networks.

This shortcoming of random graph models motivates studying deterministic properties of scale-free models, as these deterministic properties can be checked for real-world networks. To describe the properties of scale-free networks without the use of random graphs, Aiello et al [2] define (α, β)-Power Law Graphs. The problem of this model is that it essentially demands a perfect power law degree distribution, whereas the degree distributions of real networks normally exhibit slight deviations from power-laws. Therefore, (α, β)-Power Law Graphs are too constrained and do not capture most real networks.

To allow for those deviations in the degree distribution Brach et al. [11] define buckets containing nodes of degrees $\left[2^i, 2^{i+1}\right)$. If the number of nodes in each bucket is at most as high as for a power-law degree sequence, a network is said to be *power-law bounded*, which we denote as a network with property **PLB-U**. They also define the property of *PLB neighborhoods*: A network has PLB neighborhoods if every node of degree k has at most as many neighbors of degree at least k as if those neighbors were picked independently at random with probability proportional to their degree. This property we abbreviate as **PLB-N**. PLB-U and PLB-N allow some degrees of parameterization: Both properties assume a power law distribution with power-law exponent $\beta > 2$, a possible shift $t \geqslant 0$, and a scaling factor c_1 and c_3 respectively. A shift of t means that the number of nodes of degree k is proportional to $(k + t)^{-\beta}$. A formal definition of both properties can be found in Section 3. Brach et al. [11] showed experimentally that PLB-(U,N) properties hold for many real-world networks, which implies that the mentioned graph problems can be solved faster on these real-world networks than worst-case lower bounds for general graphs suggest.

2 Our Contribution

2.1 PLB properties in power-law random graph models

The PLB-(U,N) properties are designed to describe power-law graphs in a way that allows analyzing algorithms deterministically. As already mentioned, there is a multitude of random graph models [2, 7, 18, 34], which can be used to generate power-law graphs. Brach et al. [11] proved that the Erased Configuration Model [2] with a power-law degree distribution follows PLB-U and w. h. p. also PLB-N. In the Configuration Model a graph with a given degree sequence is sampled uniformly at random. This is done by generating $\deg(v)$ stubs for each node $v \in V$ and then matching these stubs independently and uniformly at random to create edges. In the Erased Configuration Model loops and multiple edges are removed in order to generate a simple graph. Since the Erased Configuration Model has a fixed degree sequence, it is relatively easy to prove the PLB-U property, but it is quite technical to prove the PLB-N property. There are other power-law random graph models, which are based on expected degree sequences, e.g. Chung-Lu Random Graphs [18]. Brach et al. argued that for showing the PLB-U property on these models, a typical concentration statement does not work, as it accumulates the additive error for each bucket. They leave it as a *challenging open question*, whether other random graph models also produce graphs with PLB-(U,N) properties with high probability[1].

The models we consider in Section 4 are Geometric Inhomogeneous Random Graphs, Hyperbolic Random Graphs, and Chung-Lu Random Graphs.

Geometric Inhomogeneous Random Graphs satisfy PLB-(U,N): Geometric Inhomogeneous Random Graphs (GIRGs) [12, 13, 33] consider an expected degree vector and an underlying geometry.

In GIRGs, all nodes draw a position uniformly at random and each edge (i, j) exists independently with a probability depending on $\frac{w_i \cdot w_j}{W}$ and the distance of i and j in the underlying geometry. We show:

Theorem 4.11 *Let G be a GIRG whose weight sequence $\vec{w}$ follows a general power-law with exponent $\beta' > 2$. Then, for all $2 < \beta < \beta'$ and $t = 0$ there are constants c_1 and c_3 such that G fulfills PLB-U and PLB-N with high probability.*

Hyperbolic Random Graphs satisfy PLB-(U,N): Hyperbolic Random Graphs (HRGs) [34] assume an underlying hyperbolic space. Each node is positioned uniformly at random in this space and connected to other nodes with a probability proportional to its hyperbolic distance to them. For Hyperbolic Random Graphs we show the following:

Theorem 4.14 *Let G be a HRG with $\alpha_H > \frac{1}{2}$. Then, G almost surely fulfills PLB-U and PLB-N with $\beta = 2\alpha_H + 1 - \eta$, $t = 0$, any constant $\eta > 0$, and some constants c_1 and c_3.*

---

[1] We say that an event E holds *w. h. p.*, if there exists a $\delta > 0$ such that $\Pr[E] \geqslant 1 - \mathcal{O}(n^{-\delta})$, and *almost surely* if it holds with probability $\Pr[E] \geqslant 1 - o(1)$.

Chung-Lu Random Graphs satisfy PLB-(U,N): Chung-Lu Random Graphs (CLRGs) [18] assume a sequence of expected degrees $w_1, w_2, \ldots, w_n$ and each edge (i, j) exists independently at random with probability $\min(1, \frac{w_i \cdot w_j}{W})$, where $W = \sum_{i=1}^n w_i$. We show the following theorem:

Theorem 4.16 *Let G be a CLRG whose weight sequence $\vec{w}$ follows a general power-law with exponent $\beta' > 2$. Then, for all $2 < \beta < \beta'$ and $t = 0$ there are constants c_1 and c_3 such that G fulfills PLB-U and PLB-N with high probability.*

2.2 Algorithmic Results

The above results imply that all results of Brach et al. [11] also hold w. h. p. for Geometric Inhomogeneous Random Graphs and Chung-Lu Random Graphs and almost surely for Hyperbolic Random Graphs. Therefore, the problems transitive closure, maximum matching, determinant, PageRank, matrix inverse, counting triangles and maximum clique have faster algorithms on Chung-Lu and Geometric Inhomogeneous Random Graphs w. h. p. and on Hyperbolic Random Graphs almost surely.

In this work we additionally consider the three classical NP-complete problems MINIMUM DOMINATING SET(MDS), MAXIMUM INDEPENDENT SET(MIS) and MINIMUM VERTEX COVER(MVC) on PLB-U networks. For the first two problems, positive results are already known for (α, β)-Power Law Graphs, which are a special case of graphs with the PLB-U property and an additional power law lower bound on the degree distribution (PLB-L). Note that this deterministic graph class is much more restrictive and does *not* cover typical real-world graphs. On the contrary, our positive results only assume the PLB-U property. Our algorithmic results can therefore be applied to real-world networks after measuring the respective constants of the PLB-model. In section 5 we prove our main lemma, Lemma 5.2 (the Potential Volume Lemma). Using the Potential Volume Lemma, we prove lower bounds for the size of MDS, MIS and MVC in the order of $\Theta(n)$ on PLB-U networks with exponent $\beta > 2$. This essentially means, even taking all nodes as a solution gives a constant factor approximation. Furthermore, in Theorem 5.7 we prove that the greedy algorithm actually achieves a better constant approximation ratio for MINIMUM DOMINATING SET. The positive results from Section 5 also hold for (α, β)-Power Law Graphs.

Brach et al. [11] proved that for PLB-(U,N) networks with $\beta > 3$ finding a maximum clique is solvable in polynomial time. This result gives rise to the question whether the PLB-N property can be helpful in solving other NP-complete problems on power-law graphs in polynomial time. In Section 6 we consider the mentioned NP-Complete problems MDS, MIS and MVC and prove that these problems are APX-hard even for PLB-(U,L,N) networks with $\beta > 2$. Therefore, at least for the three problems we considered, even the PLB-N property is not enough to make those problems polynomial-time solvable. As a side product we also get a lower-bound on the approximability of the respective problems under some complexity theoretical assumptions. Since the negative results for (α, β)-Power Law Graphs imply the same non-approximability on

graphs with PLB-(U,L), we only consider simple graphs with PLB-(U,L,N) in Section 6.

Finally, we show that all three problems are in MAX-SNP for graphs with PLB-U and $\beta > 2$. This implies that, if we reduce any of those problems to MAX 3-SAT, there exists an $fpt-$algorithm where the parameter is the maximum number of satisfiable clauses minus a lower-bound on this number, which is linear in the total number of clauses. This parameter can be considerably smaller than the solution size of the original problem.

Dominating Set: Given a Graph $G = (V, E)$, a MINIMUM DOMINATING SET (MDS) is a subset $S \subseteq V$ of minimum size such that for each $v \in V$ either v or a neighbor of v is in S. MDS cannot be approximated within a factor of $(1 - \varepsilon) \ln|V|$ for any $\varepsilon > 0$ [25] unless $\mathsf{NP} \subseteq \mathsf{DTIME}(|V|^{\log\log|V|})$ and not to within a factor of $\ln\Delta - c\ln\ln\Delta$ for some $c > 0$ [16] unless $\mathsf{P} = \mathsf{NP}$, although a simple greedy algorithm achieves an approximation ratio of $1 + \ln\Delta$ [32]. We also know that even for sparse graphs, MDS cannot be approximated within a factor of $o(\ln(n))$, since we could have a graph with a star of $n - \sqrt{n}$ nodes to which an arbitrary graph of the $\sqrt{n}$ remaining nodes is attached [36]. Furthermore, if we parameterize DOMINATING SET with the size of the solution as a parameter, it is $\mathsf{W}[2]-$complete [21].

MDS has already been studied in the context of (α, β)-Power Law Graphs. Ferrante, Pandurangan, and Park [26] showed that the problem remains NP-hard for $\beta > 0$. Shen, Nguyen, Xuan, and Thai [46] proved that there is no $\left(1 + \frac{1}{3120\zeta(\beta)3^\beta}\right)$-approximation[2] for $\beta > 1$ unless $\mathsf{P} = \mathsf{NP}$. They also showed that the greedy algorithm achieves a constant approximation factor for $\beta > 2$, showing that in this case the problem is APX-hard. Gast, Hauptmann, and Karpinski [29] also proved a logarithmic lower bound on the approximation factor when $\beta \leqslant 2$.

For graphs with the PLB-U property and power law exponent $\beta > 2$ we will show a lower bound on the size of the minimum dominating set in the range of $\Theta(n)$, which already gives us a constant factor approximation by taking all nodes. This also means that any brute-force algorithm which runs in exponential time is in FPT when we take the solution size as a parameter.

In contrast to (α, β)-Power Law Graphs the PLB-U property captures a wide range of real networks, making it possible to transfer our results to them. All our upper bounds are in terms of the following two expressions, which depend on the parameters c_1, β and t of the PLB-U property (cf. Definition 3.1):

$$a_{\beta,t} := \left(1 + \frac{\beta - 1}{\beta - 2} \frac{1}{1 - \left(\frac{t+2}{t+1}\right)^{1-\beta}}\right) \text{ and}$$

$$b_{c_1,\beta,t} := \left(c_1 \frac{\beta - 1}{\beta - 2} \cdot 2^\beta \cdot (t+1)^{\beta-1}\right)^{\frac{1}{\beta-2}}.$$

---

[2] Throughout the paper, ζ refers to the Riemann zeta function.

In the rest of the paper we assume the parameters c_1, β and t to be constants, which implies that $a_{\beta,t}$ and $b_{c_1,\beta,t}$ are constant as well.

Theorem 5.3 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the minimum dominating set is of size at least*

$$\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)^{-1} n = \Theta(n).$$

Furthermore, we will show that the greedy algorithm actually achieves a lower approximation factor than the one we get from the bound in Theorem 5.3 (see Figure 3 for a comparison).

Theorem 5.7 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the classical greedy algorithm for MINIMUM DOMINATING SET (cf. [22]) has an approximation factor of at most*

$$\log_3(5) \cdot a_{\beta,t} \ln\left(b_{c_1,\beta,t} + 1\right) + 1 = \Theta(1).$$

Note that in networks with PLB-U the maximum degree can be $\Delta = \Theta(n^{\frac{1}{\beta-1}})$. That means the simple bound for the greedy algorithm gives us only an approximation factor of $\ln(\Delta + 1) = \Theta(\log n)$.

In MINIMUM CONNECTED DOMINATING SET we are looking for a smallest dominating set S with the extra property that the induced subgraph of S in G is connected. For this related problem we prove the following constant approximation factor for the greedy algorithm introduced by Ruan et al. [44].

Theorem 5.8 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the greedy algorithm for MINIMUM CONNECTED DOMINATING SET (cf. [44]) has an approximation factor of at most*

$$2 + \ln\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right) = \Theta(1).$$

Furthermore, we show that MINIMUM DOMINATING SET remains APX-hard on networks with PLB-U and $\beta > 2$, even with the PLB-L and PLB-N property. Finally, we prove that on networks with PLB-U and $\beta > 2$, MINIMUM DOMINATING SET is in MAX SNP.

Independent Set: For a graph $G = (V, E)$, MAXIMUM INDEPENDENT SET (MIS) consists of finding a subset $S \subseteq V$ of maximum size, such that no two different vertices $u, v \in S$ are connected by an edge. MIS cannot be approximated within a factor of Δ^ε for some $\varepsilon > 0$ unless $\mathsf{P} = \mathsf{NP}$ [6], although a simple greedy algorithm achieves an approximation factor of $\frac{\Delta+2}{3}$ [30]. We also know from Turán's Theorem that every graph with an average degree of $\bar{d}$ has a maximum independent set of size at least $\frac{n}{\bar{d}+1}$. This lower bound can

---

[3] Both these results hold for the naïve greedy algorithm based on node degrees, not the one based on constructing a maximal matching, which achieves an approximation factor of 2 on general graphs.

Problem	General Graph		Graphs with PLB-U
MDS	$\mathcal{O}(\ln \Delta)$	[32]	$\Theta_n(1)$ [Theorem 5.7]
MVC	$\mathcal{O}(\ln \Delta)^3$	[Theorem 5.15]	$\Theta_n(1)$ [Corollary 5.17]
MIS	$\mathcal{O}(\Delta)$	[22]	$\Theta_n(1)$ [Corollary 5.14]
MCDS	$\mathcal{O}(\ln \Delta)$	[44]	$\Theta_n(1)$ [Theorem 5.8]

Table 1 Comparison of the approximation ratios achieved by greedy algorithms on networks with an upper bound on the power-law degree distribution (PLB-U) and exponent $\beta > 2$ and on general graphs. While on general graphs, greedy achieves only a logarithmic or polynomial approximation in the maximum degree, greedy achieves a constant-factor-approximation on graphs with PLB-U and $\beta > 2$.

already be achieved by the same greedy algorithm [30, Theorem 1]. When we consider parameterized INDEPENDENT SET with solution size as the parameter, it is W[1]-complete [21].

MIS has also been studied in the context of (α, β)-Power Law Graphs. Ferrante et al. [26] showed that the problem remains NP-hard for $\beta > 0$. Shen et al. [46] proved that for $\beta > 1$ there is no $\left(1 + \frac{1}{1120\zeta(\beta)3^\beta} - \varepsilon\right)$-approximation unless P = NP and Hauptmann and Karpinski [31] gave the first non-constant bound on the approximation ratio of MIS for $\beta \leqslant 1$.

Since the PLB-U property with $\beta > 2$ induces a constant average degree, the greedy algorithm already gives us a constant approximation factor for MAXIMUM INDEPENDENT SET on networks with these properties. Although we can not give better bounds for the maximum independent set, Theorem 5.3 immediately implies a lower bound for the size of *all maximal independent sets*.

Theorem 5.13 *In a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, every maximal independent set is of size at least*

$$\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)^{-1} n = \Theta(n).$$

It is easy to see that these lower bounds do not hold in sparse graphs in general, since in a star the center node also constitutes a maximal independent set.

Furthermore, we show that MAXIMUM INDEPENDENT SET remains APX-hard in networks with PLB-U and $\beta > 2$, even with the PLB-L and PLB-N property. Finally, we prove that on networks with PLB-U and $\beta > 2$, MAXIMUM INDEPENDENT SET is also in MAX SNP.

Vertex Cover: Given a graph $G = (V, E)$, MINIMUM VERTEX COVER (MVC) consists of finding a subset $S \subseteq V$ of minimum size such that each edge $e \in E$ is incident to at least one node from S. MVC cannot be approximated within a factor of $10\sqrt{5} - 21 \approx 1.3606$ unless P=NP, whereas the simple algorithm which greedily constructs a maximal matching achieves an

Problem	General Graph		Graph with PLB-(U,L,N)
MDS	$\Omega(\ln \Delta)$	[16]	$1 + \Omega(1)$ [Theorem 6.8]
MVC	$\geqslant 1.3606$	[20]	$1 + \Omega(1)$ [Theorem 6.12]
MIS	$\Omega(\mathsf{poly}(\Delta))$	[6]	$1 + \Omega(1)$ [Theorem 6.10]

Table 2 Comparison of the approximation lower bounds for polynomial-time algorithms (assuming $\mathsf{P} \neq \mathsf{NP}$) on networks with an upper (PLB-U) and lower (PLB-L) bound on the power-law degree distribution and with PLB neighborhoods (PLB-N) with the approximation lower bounds on general graphs. Even with the additional properties of PLB-L and PLB-N the problems on graphs with PLB-U remain APX-hard, i.e. these problems cannot admit a PTAS. The actual lower bounds for each problem can be found in the respective theorem. Here $\Omega(1)$ hides the PLB-L parameters β, t and c_2.

approximation ratio of 2 [41]. Unfortunately, the greedy algorithm based on node degrees only achieves an approximation factor of $\ln \Delta$ (Theorem 5.15).

Minimum Vertex Cover has also been studied in the context of (α, β)-Power Law Graphs. Shen et al. [46] proved that there is no PTAS for $\beta > 1$ under the Unique Games Conjecture.

We can show that in networks with PLB-U and without isolated vertices the minimum vertex cover has to have a size of at least $\Theta(n)$. This follows immediately from Theorem 5.3, since in a graph without isolated nodes every vertex cover is also a dominating set:

Theorem 5.16 *In a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the minimum vertex cover is of size at least*

$$(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)^{-1} n.$$

Also, we show that Minimum Vertex Cover remains APX-hard in networks with PLB-U and $\beta > 2$, even with the PLB-L and PLB-N property. After that we prove that on networks with PLB-U and $\beta > 2$, Minimum Vertex Cover is in MAX SNP.

3 Preliminaries and Notation

We mostly consider undirected multigraphs $G = (V, E)$ without loops, where V denotes the set of vertices and E the multiset of edges with $n = |V|$. In the following we will refer to multigraphs as graphs and state explicitly if we talk about simple graphs. Throughout the paper we use $\deg(v)$ to denote the degree of node v. Furthermore, we use d_{min} and Δ to denote the minimum and maximum degree of the graph respectively. For a set $S \subseteq V$, we let $\text{VOL}(S) = \sum_{v \in S} \deg(v)$ denote the volume of S. We use b_i to denote the set of nodes $v \in V$ with $\deg(v) \in [2^i, 2^{i+1})$. For $v \in V$ we let $N(v) = \{u \in V \mid \{u, v\} \in E\}$ denote the *exclusive neighborhood of v* and we let $N^+(v) = \{u \in V \mid u = v \lor \{u, v\} \in E\}$ denote the *inclusive neighborhood of v*. Analogously, for a set $S \subseteq V$, we let $N^+(S) = \{v \in V \mid \exists u \in S : v \in N^+(u)\}$ denote the *inclusive neighborhood of S*

and we let $N(S) = N^+(S) \setminus S$ denote the *exclusive neighborhood of S*. Furthermore, for $v \in V$ we let $N^r(v) = \{u \in V \mid \text{dist}(u, v) \leqslant r\}$ denote the *r-closed neighborhood of v*, where $\text{dist}(u, v)$ denotes the length of a shortest path from u to v in G. Analogously, for a set of nodes $S \subseteq V$ we let $N^r(S) = \bigcup_{v \in S} N^r(v)$ denote the *r-closed neighborhood of S*. If not stated otherwise log denotes the logarithm of base 2 and ln denotes the natural logarithm.

Now we give a formal definition of the PLB-U, PLB-L and PLB-N properties.

Definition 3.1 (PLB-U [11]) Let G be an undirected n-vertex graph and let $c_1 > 0$ be a universal constant. We say that G is power law upper-bounded (PLB-U) for some parameters $1 < \beta = \mathcal{O}(1)$ and $t \geqslant 0$ if for every integer $d \geqslant 0$, the number of vertices v, such that $\deg(v) \in [2^d, 2^{d+1})$ is at most

$$c_1 n(t+1)^{\beta-1} \sum_{i=2^d}^{2^{d+1}-1} (i+t)^{-\beta}.$$

Definition 3.2 (PLB-L) Let G be an undirected n-vertex graph and let $c_2 > 0$ be a universal constant. We say that G is power law lower-bounded (PLB-L) for some parameters $1 < \beta = \mathcal{O}(1)$ and $t \geqslant 0$ if for every integer $\lfloor \log d_{min} \rfloor \leqslant d \leqslant \lfloor \log \Delta \rfloor$, the number of vertices v, such that $\deg(v) \in [2^d, 2^{d+1})$ is at least

$$c_2 n(t+1)^{\beta-1} \sum_{i=2^d}^{2^{d+1}-1} (i+t)^{-\beta}.$$

Since the PLB-U property alone can capture a much broader class of networks, for example empty graphs and rings, this lower-bound is important to restrict networks to real power-law networks. In the definition of PLB-L d_{min} and Δ are necessary because in real world power law networks there are no nodes of lower or higher degree, respectively.

It is also noteworthy, that not all values of c_1 (respectively c_2) are eligible, given t, β, and n. These constants have to be big (small) enough for the buckets to encompass all n nodes. If this is the case, we call c_1 (c_2) *admissable*.

Definition 3.3 (PLB-N [11]) Let G be an undirected n-vertex graph with PLB-U for some parameters $1 < \beta = \mathcal{O}(1)$ and $t \geqslant 0$. We say that G has PLB neighborhoods (PLB-N) if for every vertex v of degree k, the number of neighbors of v of degree at least k is at most $c_3 \max \left(\log n, (t+1)^{\beta-2} k \sum_{i=k}^{n-1} i(i+t)^{-\beta} \right)$ for some universal constant $c_3 > 0$.

Brach et al. [11] assume that PLB-U and PLB-N for the same graph have the same values of t and β. This makes sense, since both bounds describe the same (power-law) degree sequence. Henceforth, in the rest of this paper, we will assume that PLB-U, PLB-N, and PLB-L for the same graph also have the same values of t and β. How well the degree distributions of real networks and
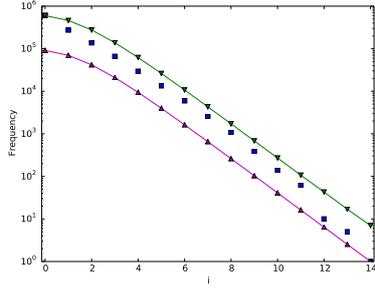
(a) Youtube Network fitted with $\beta = 2.34$, $c_1 = 1.46$, $c_2 = 0.20$, and $t = 1.76$.
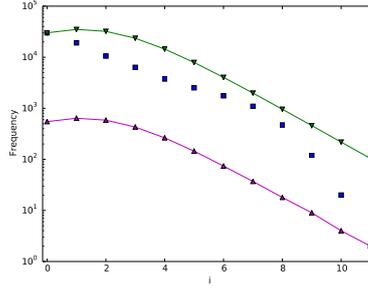
(b) Epinion Network fitted with $\beta = 2.08$, $c_1 = 1.92$, $c_2 = 0.03$, and $t = 3.84$.
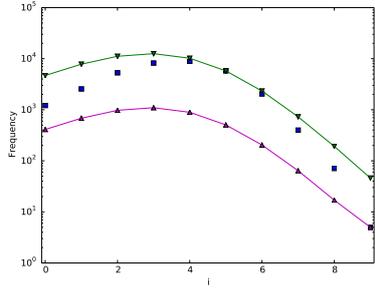
(c) HepPh fitted with $\beta = 3.22$, $c_1 = 3.43$, $c_2 = 0.30$, and $t = 24.14$.
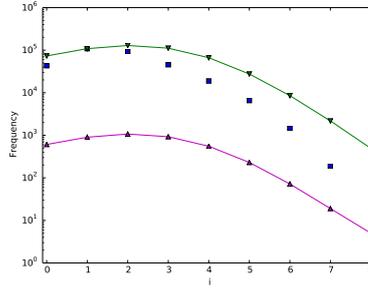
(d) DBLP Network fitted with $\beta = 3.34$, $c_1 = 3.61$, $c_2 = 0.03$, and $t = 14.64$.
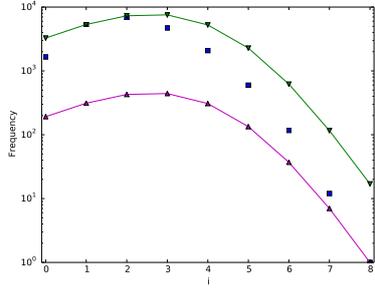
(e) CondMat Network fitted with $\beta = 4.28$, $c_1 = 4.62$, $c_2 = 0.27$, and $t = 29.12$.
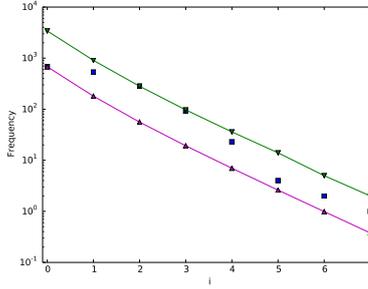
(f) Hyperbolic Random Graph fitted with $\beta = 2.38$, $c_1 = 2.11$, $c_2 = 0.41$, and $t = 0$.

Fig. 1 Total number of nodes with degrees in ranges $\left[2^i, 2^{i+1}\right)$ for $i \geqslant 0$ for the giant components of some real-world networks from the SNAP dataset [37] and a Hyperbolic Random Graph. Upper bounds (▼) fitted to function PLB-U (see Definition 3.1). Lower bounds (▲) fitted to function PLB-L (see Definition 3.2) with t and β the same as for PLB-U. As can be seen the constants c_1, c_2, β, and t of PLB-U and PLB-L are rather small. Additionally, in most cases c_1 and c_2 are close together, which indicates a good fit. Furthemore, our constants differ from those by Brach et al. [11], since they first converted their networks to directed graphs, which is not necessary in our case.

randomly generated power-law graphs fit into the PLB-U and PLB-L bounds can be seen in Figure 1.

Throughout the paper we will also make repeated use of the following Lemma, which is a more precise version of [11, Lemma 2.2]. It will help us relate power-law bounds to the bucketed bounds of the PLB properties.

Lemma 3.4 *Let* $1 \leqslant a \leqslant b/2$, *for* $a, b \in \mathbb{N}$, *and let* $c > 0$ *be a constant. Then*

$$a^{-c} \leqslant \frac{c}{1 - 2^{-c}} \sum_{i=a}^{b-1} i^{-c-1}.$$

Proof

$$\sum_{i=a}^{b-1} i^{-c-1} \geqslant \int_a^b x^{-c-1}\,\mathrm{d}x = \frac{1}{c}\left(a^{-c} - b^{-c}\right) \geqslant \frac{1 - 2^{-c}}{c} \cdot a^{-c}.$$

$$\square$$

Furthermore, we will use the following standard Chernoff Bounds (cf. [23, Theorem 1.1]) to show that our random models generate graphs with PLB-U and PLB-N.

Theorem 3.5 *Let* $X := \sum_{i \in [n]} X_i$, *where* X_i *for* $i \in [n]$ *are independently distributed in* $[0, 1]$. *Then, for* $0 < \varepsilon < 1$,

$$\Pr\left(X > (1 + \varepsilon)\mathbb{E}\left[X\right]\right) \leqslant \exp\left(-\frac{\varepsilon^2}{3}\mathbb{E}\left[X\right]\right)$$

and

$$\Pr\left(X < (1 - \varepsilon)\mathbb{E}\left[X\right]\right) \leqslant \exp\left(-\frac{\varepsilon^2}{2}\mathbb{E}\left[X\right]\right).$$

If $t > 2 \cdot e \cdot \mathbb{E}\left[X\right]$, *then*

$$\Pr\left(X > t\right) \leqslant 2^{-t}.$$

4 Power-Law Random Graphs and the PLB properties

In this section we analyze some well-known power law random graph models and prove that w. h. p. or *almost surely* graphs generated by these models have PLB-U and PLB-N properties. We consider (α, β)-Power Law Graphs, Chung-Lu Random Graphs, Geometric Inhomogeneous Random Graphs, and Hyperbolic Random Graphs, because they are common models and rather easy to analyze. Furthermore, they assume independence or some geometrically implied sparseness of edges, which is important for establishing the PLB-N property.

4.1 (α, β)-Power Law Graph

First, we consider (α, β)-Power Law Graphs. Note that (α, β)-Power Law Graphs are no random graph model. Instead, they are classes of graphs whose degree distributions follow a power-law with scaling e^α and exponent β. We show that this already ensures the PLB-U property. We will also see that they satisfy PLB-N if they are generated with the Erased Configuration Model. This follows with a result by Brach et al [11].

Formally, (α, β)-Power Law Graphs are defined as follows.

Definition 4.1 ((α, β)-Power Law Graph [3]) An (α, β)-Power Law Graph is an undirected multigraph with the following degree distribution depending on two given values α and β. For $1 \leqslant i \leqslant \Delta = \left\lfloor e^{\alpha/\beta} \right\rfloor$ there are $y_i = \left\lfloor \frac{e^\alpha}{i^\beta} \right\rfloor$ nodes of degree i.

Intuitively those graphs already satisfy PLB-U by definition. The following theorem confirms this intuition. Remember that throughout the paper ζ denotes the Riemann zeta function.

Theorem 4.2 *The (α, β)-Power Law Graph with $\beta > 1$ has the PLB-U property with $c_1 = \frac{1}{\zeta(\beta)}$, $t = 0$ and exponent β.*

Proof It holds that the number of nodes of degree between 2^d and $2^{d+1} - 1$ is at most

$$e^\alpha \sum_{i=2^d}^{2^{d+1}-1} i^{-\beta} \leqslant \frac{n}{\zeta(\beta)} \sum_{i=2^d}^{2^{d+1}-1} i^{-\beta}$$

due to the definition of the degree distribution and the fact that $n = \lfloor \zeta(\beta) e^\alpha \rfloor$ for $\beta > 1$. □

Since the degree sequence of those graphs follow a power-law, we can show that PLB-L holds as well.

Theorem 4.3 *The (α, β)-Power Law Graph with $\beta > 1$ has the PLB-L property with $c_1 = \frac{1}{2\zeta(\beta)}$, $t = 0$ and exponent β.*

Proof The number of nodes of degree i is exactly $\left\lfloor \frac{e^\alpha}{i^\beta} \right\rfloor$. Since $i \leqslant \left\lfloor e^{\alpha/\beta} \right\rfloor$, this number is at least one. Therefore $\left\lfloor \frac{e^\alpha}{i^\beta} \right\rfloor \geqslant \frac{1}{2} \frac{e^\alpha}{i^\beta}$. It now holds that the number of nodes of degree between 2^d and $2^{d+1} - 1$ is at least

$$\frac{e^\alpha}{2} \sum_{i=2^d}^{2^{d+1}-1} i^{-\beta} = \frac{n}{2\zeta(\beta)} \sum_{i=2^d}^{2^{d+1}-1} i^{-\beta}$$

due to the definition of the degree distribution and the fact that $n = \zeta(\beta) e^\alpha$ for $\beta > 1$. □

Since (α, β)-Power Law Graphs describe classes of graphs with given degree distributions, PLB-N is not satisfied automatically. However, one can sample those graphs randomly with the Erased Configuration Model [2] to guarantee PLB-N with high probability. The Configuration Model gets a degree sequence as input and randomly generates a multigraph with this sequence. This is done by creating $\deg(u)$ many stubs for each node u and then connecting stubs uniformly at random. In the Erased Configuration Model loops and multi-edges are erased after this process in order to generate simple graphs. Brach et al [11] proved that random networks created by the Erased Configuration Model whose prescribed degree sequence follows PLB-U, also follow PLB-U and PLB-N with high probability. This yields the following lemma, which concludes our section on (α, β)-Power Law Graphs.

Lemma 4.4 ([11]) *A random (α, β)-Power Law Graph with $\beta > 1$ created with the Erased Configuration Model has the PLB-U and PLB-N properties with high probability.*

4.2 Geometric Inhomogeneous Random Graphs

In this section we consider the very general model of Geometric Inhomogeneous Random Graphs (GIRGs), which was introduced by Bringmann et al [12]. In this model, nodes are distributed uniformly at random on some underlying ground space. The probability of creating an edge between two nodes then depends on given weights for those nodes and on their distance in the ground space. Formally, the model is defined as follows.

Definition 4.5 (Geometric Inhomogeneous Random Graphs (GIRGs) [12, 14]) A Geometric Inhomogeneous Random Graph is a simple graph $G = (V, E)$ with the following properties. For $|V| = n$ let $w = (w_1, \cdots, w_n)$ be a sequence of positive weights. Let $W = \sum_{i=1}^{n} w_i$ be the total weight. For any vertex v, draw a point x_v uniformly at random from the d-dimensional torus $\mathbb{T}^d = \mathbb{R}^d \setminus \mathbb{Z}^d$ with $d \in \mathbb{N}^+$. We connect vertices $u \neq v$ independently with probability $p_{uv} = p_{uv}(r)$, which depends on the weights w_u, w_v and on the positions x_u, x_v, more precisely, on the distance $r = \|x_u - x_v\|$. For some fixed constant $\alpha > 1$ the edge probability is

$$p_{uv} = \Theta\left(min\left\{ \frac{1}{\|x_u - x_v\|^{\alpha d}} \left(\frac{w_u w_v}{W}\right)^{\alpha}, 1 \right\} \right).$$

The definition of GIRGs lends itself to encompass a multitude of models with a high degree of freedom regarding expected degree distributions and underlying geometries. Thus, we need additional constraints to show that this model generates instances that are power-law bounded. One straightforward condition is that the expected node degrees are power-law distributed. The following definition captures this condition formally.

Definition 4.6 (General Power-law [12]) A weight sequence $\vec{w}$ is said to follow a general power-law with exponent $\beta > 2$ if $w_{\min} := \min \{w_v \mid v \in V\} =$

$\Omega(1)$ and if there is a $\bar{w} = \bar{w}(n) \geqslant n^{\omega(1/\log\log n)}$ such that for all constants $\eta > 0$ there are $\varepsilon_1, \varepsilon_2 > 0$ with

$$\varepsilon_1 \frac{n}{w^{\beta-1+\eta}} \leqslant |\{v \in V \mid w_v \geqslant w\}| \leqslant \varepsilon_2 \frac{n}{w^{\beta-1-\eta}},$$

where the first inequality holds for all $w_{\min} \leqslant w \leqslant \bar{w}$ and the second holds for all $w \geqslant w_{\min}$.

Note that the former definition states that for any choice of $\eta > 0$ one can find constants c_1 and c_2 with the desired properties. However, for PLB-L, PLB-U, and PLB-N to hold it is sufficient to choose a fixed η and constants ε_1 and ε_2 that satisfy the property. Another property implied by the general power-law is that the maximum degree is $\Delta = \mathcal{O}\left(n^{1/(\beta-\eta-1)}\right)$. We will use this property in the proof of Theorem 4.11.

We are now going to prove that GIRGs fulfill PLB-U and PLB-N. For this we need the following theorem and auxiliary lemmas by Bringmann et al. [13].

Theorem 4.7 ([13]) *Let G be a GIRG with a weight sequence that follows a general power-law with exponent β and average degree $\Theta(1)$. Then, with high probability the degree sequence of G follows a general power law with exponent β and average degree $\Theta(1)$, i.e for all constants $\eta > 0$ there exist constants $\varepsilon_3, \varepsilon_4 > 0$ such that w. h. p.*

$$\varepsilon_3 \frac{n}{k^{\beta-1+\eta}} \leqslant |\{v \in V | deg(v) \geqslant k\}| \leqslant \varepsilon_4 \frac{n}{k^{\beta-1-\eta}},$$

where the first inequality holds for all $1 \leqslant d \leqslant \bar{w}$ and the second holds for all $d \geqslant 1$.

The following three lemmas are necessary to prove Theorem 4.11. Lemma 4.8 states that the marginal edge probability between two nodes u and v is essentially $\min\left\{1, \frac{w_u w_v}{W}\right\}$. Furthermore, even conditioned on a node's position x_u, all edges between u and other nodes are present independently.

Lemma 4.8 ([13]) *Fix $u \in [n]$ and $x_u \in \mathbb{T}^d$. All edges $\{u, v\}$, $u \neq v$, are independently present with probability*

$$\Pr[u \sim v \mid x_u] = \Theta(\Pr[u \sim v]) = \Theta(\min\left\{1, \frac{w_u w_v}{W}\right\}).$$

In the proof of Theorem 4.11 we will only use the marginal edge probability from the former lemma. Lemma 4.9 bounds the expected node degrees asymptotically.

Lemma 4.9 ([13]) *For any $v \in [n]$ in a Geometric Inhomogeneous Random Graph, we have*

$$\mathbb{E}[deg(v)] = \Theta(w_v).$$

The former two lemmas imply that we can use standard Chernoff bounds to bound node degrees, but we also need the following auxiliary lemma. It bounds the expected volume of nodes with given maximum or minimum weights.

Lemma 4.10 ([13]) *Let $\vec{w}$ be a general power-law weight sequence with exponent β and let $W_{\geqslant w} = \sum_{\vec{w}_v : \vec{w}_v \geqslant w} \vec{w}_v$ and $W_{\leqslant w} = \sum_{\vec{w}_v : \vec{w}_v \leqslant w} \vec{w}_v$. Then the total weight satisfies $W = \Theta(n)$. Moreover, for all sufficiently small $\eta > 0$,*

(i) $W_{\geqslant w} = \mathcal{O}(nw^{2-\beta+\eta})$ for all $w \geqslant w_{\min}$,
(ii) $W_{\geqslant w} = \Omega(nw^{2-\beta-\eta})$ for all $w_{\min} \leqslant w \leqslant \bar{w}$,
(iii) $W_{\leqslant w} = \mathcal{O}(n)$ for all w, and
(iv) $W_{\leqslant w} = \Omega(n)$ for all $w = \omega(1)$.

We are now ready to prove our first main theorem: For GIRGs whose weight sequence follows a general power-law with exponent β', we can always find constants c_1 and c_2 such that PLB-U and PLB-N are satisfied for $t = 0$ and *any* power law exponent $2 < \beta < \beta'$. Intuitively, this holds, since PLB-U and PLB-N only demand upper bounds on the number of nodes in a bucket or the neighborhood of a node. By decreasing the power-law exponent in PLB-U and PLB-N, these upper bounds only get less restrictive.

The proof works as follows. First, we show that Theorem 4.7 implies PLB-U. Second, we show PLB-N by bounding the ranges of node degrees depending on weights. We show that only nodes of sufficiently high weights can have a degree of at least k. It now suffices to consider those nodes with high weights as potential neighbors of a degree-k node. Since the edges between a degree-k node and its potential neighbors are drawn independently, we can use a Chernoff bound to show that the number of neighbors of degree at least k is concentrated around its expected value. All these statements hold with high probability. Thus, we can simply collect the error probabilities without considering dependencies. This shows PLB-N.

Theorem 4.11 *Let G be a GIRG whose weight sequence $\vec{w}$ follows a general power-law with exponent $\beta' > 2$. Then, for all $2 < \beta < \beta'$ and $t = 0$ there are constants c_1 and c_3 such that G fulfills PLB-U and PLB-N with high probability.*

Proof First, we show that G fulfills PLB-U with high probability. Let $k = 2^d$. It now holds that

$$|\{v \in V \mid deg(v) \geqslant k\}| \leqslant \varepsilon_4 \frac{n}{k^{\beta'-1-\eta}}$$

$$\leqslant \varepsilon_4 n \frac{\beta'-1-\eta}{1 - 2^{-\beta'+1+\eta}} \sum_{i=k}^{2k-1} i^{-\beta'+\eta}$$

due to Theorem 4.7 and Lemma 3.4. This means that G has the PLB-U property with $\beta = \beta' - \eta$, $t = 0$ and $c_1 = \varepsilon_4 \frac{\beta'-1-\eta}{1-2^{-\beta'+1+\eta}}$.

Now we show that G also fulfills PLB-N with high probability. To this end, we consider a node v with weight w_v. We first bound the range into which the degree $deg(v)$ of v can fall with high probability. We pessimistically assume that $deg(v)$ takes its lower bound, because then the number of possible neighbors of degree at least $deg(v)$ is maximized. Further, we assume that all other nodes' degrees take their respective upper bounds. Within these bounds, only nodes with high enough weights can reach a degree of at least $deg(v)$.

We call these nodes *potential neighbors of* v. Finally, we bound the number of edges between v and its potential neighbors. This can be done with a standard Chernoff bound, since all edges between v and its potential neighbors are drawn independently.

Due to Lemma 4.8 we can use standard Chernoff bounds as stated in Theorem 3.5 to bound the degrees of nodes. According to Lemma 4.9 there are constants $c_7, c_8 > 0$ such that

$$c_7 \cdot w_v \leqslant \mathbb{E}[\deg(v)] \leqslant c_8 \cdot w_v$$

holds for all $v \in V$. Let c be an appropriately chosen constant. For a node $v \in V$ with $w_v \geqslant c \ln n$ it holds that

$$\Pr\left(\deg(v) > \frac{3}{2} \cdot \mathbb{E}\left[\deg(v)\right]\right) \leqslant e^{-\frac{\mathbb{E}[\deg(v)]}{12}} \leqslant e^{-\frac{c_7 \cdot w_v}{12}} \leqslant n^{-\frac{c \cdot c_7}{12}}$$

and

$$\Pr\left(\deg(v) < \frac{1}{2} \cdot \mathbb{E}\left[\deg(v)\right]\right) \leqslant e^{-\frac{\mathbb{E}[\deg(v)]}{8}} \leqslant e^{-\frac{c_7 \cdot w_v}{8}} \leqslant n^{-\frac{c \cdot c_7}{8}}.$$

For a sufficiently large constant c it holds w. h. p. that

$$\frac{1}{2} \cdot c_7 \cdot w_v \leqslant \frac{1}{2} \cdot \mathbb{E}\left[\deg(v)\right] \leqslant \deg(v) \leqslant \frac{3}{2} \cdot \mathbb{E}\left[\deg(v)\right] \leqslant \frac{3}{2} \cdot c_8 \cdot w_v. \quad (1)$$

Thus, the degrees of nodes with weights at least logarithmic in n are concentrated around their weights. However, this does not hold for nodes with smaller weights. Therefore, we will consider them separately. First, we show that nodes with smaller weights cannot reach much higher degrees than $c \cdot \ln n$. Thus, for nodes of high weight and high degree they do not have to be considered as neighbors. This is because PLB-N is only concerned with neighbors of same or higher degree. Second, we show that nodes with smaller weights also comply with PLB-N. Remember that PLB-N allows a node of degree k to have $c_3 \cdot \log n$ neighbors of degree at least k. Since nodes with small weights of at most $c \cdot \ln n$ have at most $\mathcal{O}(\log n)$ neighbors in total with high probability, this property holds.

Let us now consider nodes $v \in V$ with low weights $w_v < c \ln n$. Due to the third statement in Theorem 3.5 it holds that

$$\Pr\left(\deg(v) > 2e \cdot c_8 \cdot c \ln n\right) \leqslant 2^{-2e \cdot c_8 \cdot c \ln n} = n^{-\frac{2e \cdot c \cdot c_8}{\log_2(e)}},$$

since $2e \cdot c_8 \cdot c \ln n > 2e \cdot c_8 \cdot w_v \geqslant 2e \cdot \mathbb{E}\left[\deg(v)\right]$. If we choose c sufficiently large, it holds w. h. p. that the degrees of these nodes are at most $2e \cdot c_8 \cdot c \ln n = \mathcal{O}(\log n)$. Thus, nodes with low weights already comply with the bound from PLB-N. Also, $2e \cdot c_8 \cdot c \ln n$ is smaller than the lower bound from inequality (1) for nodes with weight $w_u \geqslant 4e \frac{c_8}{c_7} c \ln n$.

Therefore, with high probability no node of weight $w_u \geqslant 4e \frac{c_8}{c_7} c \ln n$ can have a node with weight $w_v < c \ln n$ as a neighbor of same or higher degree.

Before we focus on nodes with high weight $w_u \geqslant 4e\frac{c_8}{c_7}c\ln n$, we still have to consider nodes $v \in V$ with intermediate weights $c\ln n \leqslant w_v < 4e\frac{c_8}{c_7}c\ln n$. Since the degrees of those nodes are concentrated around w_v, it holds that $\deg(v) \leqslant 6e\frac{c_8^2}{c_7}c\ln n = \mathcal{O}(\log n)$ w. h. p. due to inequality (1). This also complies with the bounds from PLB-N.

It remains to show that PLB-N also holds for nodes $v \in V$ with high weights $w_v \geqslant 4e\frac{c_8}{c_7}c\ln n$. Again, we can assume $\deg(v) \geqslant \frac{1}{2}c_7 \cdot w_v \geqslant 2e \cdot c_8 \cdot c\ln n$. As we have seen, no node u with $w_u < c\ln n$ can reach a degree of $\deg(v)$ with high probability. That means, the only nodes that can reach a degree of at least $\deg(v)$ w. h. p. are those with

$$w_u \geqslant \frac{1}{3} \cdot \frac{c_7}{c_8} \cdot w_v =: \hat{w}$$

due to inequality (1). These are the potential neighbors of v with degree at least $\deg(v)$. Let X be the number of edges between v and these potential neighbors. Now it holds that

$$\mathbb{E}\left[X\right] = \Theta\left(\frac{w_v}{W} \cdot W_{\geqslant \hat{w}}\right) \leqslant \mathcal{O}(w_v \cdot \hat{w}^{2-\beta'+\eta}) = \mathcal{O}(w_v^{3-\beta'+\eta})$$

due to Lemma 4.8 and Lemma 4.10. We can assume that the expected value is at most $c_9 \cdot w_v^{3-\beta'+\eta}$. Again, we can use the Chernoff bounds from Theorem 3.5 to bound the number of these edges. If $c_9 \cdot w_v^{3-\beta'+\eta} < c\ln n$ it holds that

$$\Pr\left(X > 2 \cdot e \cdot c\ln n\right) \leqslant 2^{2 \cdot e \cdot c\ln n}.$$

In that case the number of neighbors with degree at least $\deg(v)$ is at most $2 \cdot e \cdot c\ln n$ with high probability. As in the case of low weight nodes, this complies with PLB-N. If $c_9 \cdot w_v^{3-\beta'+\eta} \geqslant c\ln n$ it holds that

$$\Pr\left(X > \frac{3}{2} \cdot c_9 \cdot w_v^{3-\beta'+\eta}\right) \leqslant e^{-\frac{c_9 \cdot w_v^{3-\beta'+\eta}}{12}} \leqslant n^{-\frac{c \cdot c_9}{12}}.$$

This is a high probability if c is chosen appropriately high. It now holds w. h. p. that

$$X \leqslant \frac{3}{2} \cdot c_9 \cdot w_v^{3-\beta'+\eta} \leqslant \frac{4 \cdot c_9}{9 \cdot c_8^{3-\beta'+\eta}} \cdot \deg(v)^{3-\beta'+\eta} = c_{10} \cdot \deg(v)^{3-\beta'+\eta}$$

due to inequality (1). In order to get an expression that resembles PLB-N, we use Lemma 3.4. Although $\deg(v)$ is a random variable, for all realizations $\deg(v) = k$ within our high probability bounds, we can apply the lemma with $a = k$, $b = n$, and $c = \beta' - \eta - 2$. The requirement $1 \geqslant a \geqslant b/2$ holds, since $1 \geqslant k$ holds due to inequality (1) and $a \leqslant \Delta = \mathcal{O}\left(n^{1/(\beta'-\eta-1)}\right) = o(n)$ is implied by the general power law. Lemma 3.4 now yields

$$\deg(v)^{2-\beta'+\eta} = \mathcal{O}\left(\sum_{i=\deg(v)}^{n-1} i^{1-\beta'+\eta}\right).$$

This implies that the number of neighbors of v with degree at least $\deg(v)$ is at most $\mathcal{O}\left(\deg(v) \sum_{i=\deg(v)}^{n-1} i^{1-\beta'+\eta}\right)$, which is at most

$$c_3 \cdot (t+1)^{\beta'-\eta-1} \deg(v) \sum_{i=\deg(v)}^{n-1} i(i+t)^{-\beta'+\eta}$$

for $\beta = \beta' - \eta$, a suitable constant c_3 and $t = 0$. Thus, in all cases PLB-N holds as desired. □

We have proven that GIRGs with a power-law weight sequence generate power-law bounded graphs with high probability. There are two other random graph models which are highly related to GIRGs and for which we can show PLB-(U,N) as well. First, Hyperbolic Random Graphs, which were shown to be a special case of GIRGs by Bringmann et al [12]. Second, Chung-Lu Random Graphs, which generate random instances in a very similar way to Geometric Inhomogeneous Random Graphs, but without an underlying geometry. Therefore, we consider both of these models as special cases of GIRGs and analyze them in the following subsections.

4.2.1 Hyperbolic Random Graphs

In this section we show PLB-U and PLB-N in Hyperbolic Random Graphs (HRGs). In HRGs, nodes are uniformly distributed on the hyperbolic plane. Then, two nodes are connected with probability proportional to their hyperbolic distance. The following definition formalizes the model.

Definition 4.12 (Hyperbolic Random Graph (HRG) [34]) Let $\alpha_H > 0$, $C_H \in \mathbb{R}$, $T_H > 0$, $n \in \mathbb{N}$ and $R = 2\log n + C_H$. The Hyperbolic Random Graph $G_{\alpha_H, C_H, T_H}(n)$ is a simple graph with vertex set $V = [n]$ and the following properties:

- Every vertex $v \in [n]$ draws coordinates (r_v, ϕ_v) independently at random, where the angle ϕ_v is chosen uniformly at random in $[0, 2\pi)$ and the radius $r_v \in [0, R]$ is random according to density $f(r) = \frac{\alpha_H \sinh(\alpha_H r)}{\cosh(\alpha_H R)-1}$.
- Every potential edge $e = \{u, v\} \in \binom{[n]}{2}$ is present independently with probability
$$p_H(d(u,v)) = \left(1 + e^{\frac{d(u,v)-R}{2T_H}}\right)^{-1},$$
where $d(u,v)$ is the *hyperbolic distance* between u and v, i.e. the non-negative solution of the equation
$$\cosh\left(d(u,v)\right) = \cosh\left(r_x\right)\cosh\left(r_y\right) - \sinh\left(r_x\right)\sinh\left(r_y\right)\cos\left(\phi_x - \phi_y\right).$$

Bringmann et al [12] show that this model is a special case of GIRGs with input parameters

$$d := 1, \quad \beta := 2\alpha_H + 1, \quad \alpha := 1/T_H$$

and

$$w_v := e^{\frac{R-r_v}{2}} \quad \text{and} \quad x_v := \frac{\phi_v}{2\pi}$$

for $v \in V$. They also show this this weight sequence follows a general power law.

Lemma 4.13 ([12]) *Let $\alpha_H > \frac{1}{2}$. Then for all $\eta = \eta(n) = \omega(\frac{\log\log n}{\log n})$, with probability $1 - n^{-\Omega(n)}$ the induced weight sequence $\vec{w}$ follows a power law with parameter $\beta = 2\alpha_H + 1$.*

Thus, we can apply Theorem 4.11 to show that HRGs also satisfy PLB-U and PLB-N.

Theorem 4.14 *Let G be a HRG with $\alpha_H > \frac{1}{2}$. Then, G almost surely fulfills PLB-U and PLB-N with $\beta = 2\alpha_H + 1 - \eta$, $t = 0$, any constant $\eta > 0$, and some constants c_1 and c_3.*

4.2.2 Chung-Lu Random Graphs

In this section we consider another random graph model related to GIRGs, Chung-Lu Random Graphs (CLRGs). As in GIRGs, nodes have weights which influence the edge probability. However, there is no notion of distance between nodes. The following formal definition makes this clear.

Definition 4.15 (Chung-Lu Random Graph (CLRG) [17]) A Chung-Lu Random Graph is a simple graph $G = (V, E)$. Given a weight sequence $\mathbf{w} = (w_1, w_2, \ldots, w_n)$ the edges between nodes v_i and v_j exist independently with probability p_{ij} proportional to $\min\left(1, \frac{w_i w_j}{W}\right)$, where $W = \sum_{i=1}^{n} w_i$.

Geometric Inhomogeneous Random Graphs and Chung-Lu Random Graphs are highly related. In fact, the definition of GIRGs was originally inspired by the one for CLRGs. We can see that the same properties we used to show PLB-(U,N) for GIRGs also hold for CLRGs. Most importantly, we used the marginal edge probability between each pair of nodes in a Geometric Inhomogeneous Random Graph. However, this marginal probability is exactly the edge probability of Chung-Lu Random Graphs. From this property and the independence of edges in CLRGs one can derive all lemmas in exactly the same way, but for CLRGs instead of GIRGs. Especially, we can derive PLB-(U,N) as stated in the following theorem.

Theorem 4.16 *Let G be a CLRG whose weight sequence $\vec{w}$ follows a general power-law with exponent $\beta' > 2$. Then, for all $2 < \beta < \beta'$ and $t = 0$ there are constants c_1 and c_3 such that G fulfills PLB-U and PLB-N with high probability.*

5 Greedy Algorithms

In this section we try to understand why simple greedy algorithms work efficiently on power-law bounded graphs. The problems we consider are MINIMUM

DOMINATING SET, MINIMUM VERTEX COVER, and MAXIMUM INDEPENDENT SET. We will show that graphs with PLB-U only have dominating sets, vertex covers, and maximal independent sets of size $\Theta(n)$. This already implies that greedy algorithms for these problems achieve a constant approximation factor. However, for MDS and some variants of it, we can derive better approximation factors for greedy algorithms than the ones implied by our lower bounds on the solution size. We are able to derive these results with the help of the Potential Volume Lemma (Lemma 5.2), which we introduce in the next section.

Definition 5.1 A greedy algorithm is an α-*approximation* for problem P if it produces a solution set S with $\alpha \geqslant \frac{|S|}{|\text{OPT}|}$ if P is a minimization problem and with $\alpha \geqslant \frac{|\text{OPT}|}{|S|}$ if P is a maximization problem.

5.1 Analysis of Greedy Algorithms on PLB-U Networks

This section will be dedicated to proving our main lemma, the *Potential Volume Lemma* (Lemma 5.2).

The idea of the lemma is inspired by lower-bounding the size of a dominating set as done by Shen et al. [46] and by Gast et al. [29] in the context of (α, β)-Power-Law Graphs: In order to dominate all nodes, the volume of the dominating set has to be high enough, more precisely $\sum_{x \in S} \deg(x) + 1 \geqslant n$ for any dominating set S. However, in a power-law bounded graph at least a constant fraction of nodes are necessary to reach such a high volume.

The Potential Volume Lemma generalizes this idea and gives us upper bounds on the volume of sets of nodes depending on their size. Moreover, the lemma also works for other well-behaved non-decreasing functions on node degrees. It essentially states that for a graph with PLB-U, a set of nodes S, and a nice non-decreasing function h

$$\sum_{x \in S} h(\deg(x)) \lesssim (n/\text{VOL}(S))^{1/(\beta-2)} \cdot |S|.$$

This property can be used to derive lower bounds on the size of optimal solutions. For some greedy algorithms we can also derive upper bounds on the solution size in terms of $\sum_{x \in \text{OPT}} h(\deg(x))$. We can then use the lemma with $S = \text{OPT}$ to derive upper bounds on the greedy solution size with respect to the size and volume of an optimal solution, yielding improved approximation ratios.

The proof of the lemma can be outlined as follows. First, for any set S it holds that $\sum_{x \in S} h(\deg(x))/|S| \leqslant \sum_{x \in S'} h(\deg(x))/|S'|$ for any set S' of size at most $|S|$ that contains the $|S'|$ nodes of highest degree. This is intuitively clear if we interpret these expressions as average values of $h(\deg(\cdot))$ per node and remember that h is non-decreasing. Thus, we can achieve an upper bound by considering the average over the highest buckets which contain at most $|S|$ nodes in total. Using PLB-U and assuming that each bucket d contains as many nodes as possible and that those nodes all have maximum permitted degree $2^d - 1$, we get an upper bound on this average.
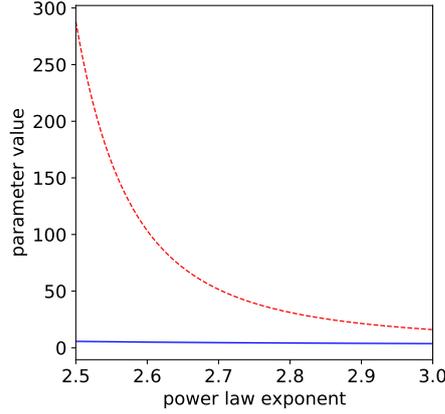
Fig. 2 Comparison of $a_{\beta,t}$ (—) and $b_{c_1,\beta,t}$ (- -) for $t = 0$, $c_1 = 1$, and $\beta \in [2.5, 3]$.

Before we state the lemma formally, remember that we use the following shorthand notations:

$$a_{\beta,t} := \left(1 + \frac{\beta-1}{\beta-2} \frac{1}{1 - \left(\frac{t+2}{t+1}\right)^{1-\beta}}\right) \text{ and}$$

$$b_{c_1,\beta,t} := \left(c_1 \frac{\beta-1}{\beta-2} \cdot 2^\beta \cdot (t+1)^{\beta-1}\right)^{\frac{1}{\beta-2}}.$$

To get a feeling of how big $a_{\beta,t}$ and $b_{c_1,\beta,t}$ are, let us consider the "ideal" case $c_1 = 1$ and $t = 0$. For $\beta \to 2$ both values approach infinity, although $a_{\beta,t}$ is orders of magnitude smaller. For $\beta \to \infty$ it holds that

$$\lim_{\beta \to \infty} a(\beta, t) = 2 \quad \text{and} \quad \lim_{\beta \to \infty} b(c_1, \beta, t) = 2 \cdot (t+1).$$

Figure 2 shows how large these values are for intermediate power-law exponents $\beta \in [2.5, 3]$.

In the lemma, we assume the graph to have a certain minimum number of nodes, which is not a big restriction. Further, we assume the feasible solution to be of size at most $|V|-1$. When applying the lemma we will always use $S = \text{OPT}$. Having $|S| = |\text{OPT}| \leqslant n - 1$ is not a big restriction, since $|\text{OPT}| = n$ is usually a degenerate case. For MDS and MIS $|\text{OPT}| = n$ would imply a graph without edges. This cannot be the case, since we forbid isolated vertices. For MVC it always holds that $|\text{OPT}| \leqslant n - 1$.

Lemma 5.2 (Potential Volume Lemma) *Let $G = (V, E)$ be a graph without loops and isolated vertices. Also, let G have the PLB-U property for some $\beta > 2$, some constant $c_1 > 0$, and some constant $t \geqslant 0$, and let*

$$|V| = n \geqslant \left(c_1 \cdot (t+1)^{\beta-1} \frac{\beta}{\beta-1} 2^{\beta-1}\right)^{\frac{1}{\beta-2}}.$$

Let $S \subseteq V$ be a feasible solution with $|S| < n$, let $g \colon \mathbb{R}^+ \to \mathbb{R}^+$ be a continuously differentiable function and let $h(x) := g(x) + C$ for some constant C such that

(i) g non-decreasing,
(ii) $g(2x) \leqslant c \cdot g(x)$ for all $x \geqslant 2$ and some constant $c > 0$,
(iii) $g'(x) \leqslant \frac{g(x)}{x}$ for all $x \geqslant 1$,

then it holds that

$$\sum_{x \in S} h(\deg(x)) \leqslant \left(c \cdot a_{\beta,t} \cdot g\left(b_{c_1,\beta,t} \cdot \left(\frac{n}{2 \cdot M(n)} \right)^{\frac{1}{\beta-2}} \right) + C \right) \cdot |S|,$$

where $M(n) \geqslant 1$ is chosen such that $\sum_{x \in S} \deg(x) \geqslant M(n)$.

Proof Without loss of generality assume the nodes of G were ordered by increasing degree, i.e. $V(G) = \{v_1, v_2, \ldots, v_n\}$ with $\deg(v_1) \leqslant \deg(v_2) \leqslant \ldots \leqslant \deg(v_n)$. Let $n' := 2^{\lfloor \log(n-1) \rfloor + 1} - 1$. This is the maximum degree of the bucket an $(n-1)$-degree node is in. For $j \in \mathbb{N}$ let

$$D(j) := c_1 \cdot n(t+1)^{\beta-1} \sum_{i=2^j}^{n'} (i+t)^{-\beta},$$

i.e. the maximum number of nodes of degree at least 2^j that G can have according to the PLB-U property, and let

$$s(\ell) := \min \{ j \in \mathbb{N} \mid D(j) \leqslant \ell \}.$$

We can interpret $s(\ell)$ as the index j of the smallest bucket, such that the total number of nodes in buckets j to $\lfloor \log(n-1) \rfloor$ is at most ℓ.

It now holds for all $\ell \in \mathbb{R}$ with $\ell \leqslant |S|$ that

$$\frac{\sum_{x \in S} h(\deg(x))}{|S|} \leqslant \frac{\sum_{i=1}^{\lfloor \ell \rfloor} h(\deg(v_i)) + (\ell - \lfloor \ell \rfloor) h(\deg(v_{\lceil \ell \rceil}))}{\ell}$$

due to the fact that g, and therefore also h, is non-decreasing. To upper bound the numerator on the right-hand side we assume that in each bucket we have the maximum (fractional) number of nodes of maximum degree, leading to at most

$$\sum_{j=s(\ell)}^{\lfloor \log(n-1) \rfloor} \left(c_1 n(t+1)^{\beta-1} \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta} \right) h(2^{j+1} - 1) + (\ell - D(s(\ell))) h(2^{s(\ell)} - 1).$$

Now let $k := s(|S|)$. It has to hold that $k \geqslant 1$, since otherwise the total upper bounds from all buckets would sum up to at most $|S| < n$. For $\ell = D(k)$ we get

$$\frac{c_1 \cdot n(t+1)^{\beta-1} \sum_{j=k}^{\lfloor \log(n-1) \rfloor} h(2^{j+1} - 1) \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta}}{D(k)} |S|$$

$$= \left(\frac{c_1 \cdot n(t+1)^{\beta-1} \sum_{j=k}^{\lfloor \log(n-1) \rfloor} g(2^{j+1} - 1) \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta}}{D(k)} + C \right) |S|,$$

$$(2)$$

since $h(x) = g(x) + C$ for some constant C and the numerator for that second term just sums up to $C \cdot D(k)$. One more note of caution: W.l.o.g. we assume $|S| \geqslant 1$. In the case of equality, we need $s(1) \leqslant \lfloor \log(n-1) \rfloor$. Otherwise the last bucket already contains more nodes than our solution, i.e. we could not take any complete bucket. For the desired inequality to hold, we must ensure

$$D(\lfloor \log(n-1) \rfloor) = c_1 \cdot n(t+1)^{\beta-1} \sum_{i=2^{\lfloor \log(n-1) \rfloor}}^{n'} (i+t)^{-\beta} \leqslant 1.$$

It holds that

$$c_1 \cdot n(t+1)^{\beta-1} \sum_{i=2^{\lfloor \log(n-1) \rfloor}}^{n'} (i+t)^{-\beta}$$

$$\leqslant c_1 \cdot n(t+1)^{\beta-1} \left((2^{\lfloor \log(n-1) \rfloor} + t)^{-\beta} + \frac{1}{\beta-1}(2^{\lfloor \log(n-1) \rfloor} + t)^{1-\beta} \right)$$

$$\leqslant c_1 \cdot n(t+1)^{\beta-1} \frac{\beta}{\beta-1} 2^{\beta-1} n^{1-\beta},$$

since $2^{\lfloor \log(n-1) \rfloor} \geqslant \frac{n}{2}$. We can see that this is at most 1 for

$$n \geqslant \left(c_1 \cdot (t+1)^{\beta-1} \frac{\beta}{\beta-1} 2^{\beta-1} \right)^{\frac{1}{\beta-2}}.$$

We start estimating equation (2) by deriving an upper bound on the numerator. Using properties (i) and (ii) we derive

$$g(2^{j+1} - 1) \leqslant g(2^{j+1}) \leqslant c \cdot g(2^j) \leqslant c \cdot g(i+t).$$

Plugging this into the numerator gives

$$c_1 \cdot n(t+1)^{\beta-1} \sum_{j=k}^{\lfloor \log(n-1) \rfloor} g(2^{j+1} - 1) \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta}$$

$$\leqslant c \cdot c_1 n(t+1)^{\beta-1} \sum_{j=k}^{\lfloor \log(n-1) \rfloor} \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta} \cdot g(i+t)$$

$$= c \cdot c_1 n(t+1)^{\beta-1} \sum_{i=2^k}^{2^{\lfloor \log(n-1) \rfloor + 1} - 1} (i+t)^{-\beta} \cdot g(i+t). \qquad (3)$$

It is easy to check that the function $g(x) \cdot x^{-\beta}$ is non-increasing by property (iii) and the fact that $\beta > 2$. We can now estimate the sum in equation (3) by an integral

$$\sum_{i=2^k}^{n'} (i+t)^{-\beta} \cdot g(i+t) \leqslant (2^k+t)^{-\beta} \cdot g(2^k+t) + \int_{x=2^k}^{n'} (x+t)^{-\beta} \cdot g(x+t) \, dx$$

$$= (2^k+t)^{-\beta} \cdot g(2^k+t) + \int_{x=2^k+t}^{n'+t} x^{-\beta} \cdot g(x) \, dx. \qquad (4)$$

Using integration by parts we get

$$
\int_{x=2^k+t}^{n'+t} x^{-\beta} \cdot g(x) \ dx
$$

$$
= \frac{1}{1-\beta} \left[x^{1-\beta} \cdot g(x) \right]_{2^k+t}^{n'+t} - \frac{1}{1-\beta} \int_{2^k+t}^{n'+t} x^{1-\beta} \cdot g'(x) \ dx
$$

$$
\leqslant \frac{1}{\beta-1} (2^k+t)^{1-\beta} g(2^k+t) + \frac{1}{\beta-1} \int_{2^k+t}^{n'+t} x^{1-\beta} g'(x) \ dx,
$$

since $\beta > 2$. Due to property (iii) it holds that $x^{1-\beta} \cdot g'(x) \leqslant x^{-\beta} \cdot g(x)$, giving us

$$
\int_{2^k+t}^{n'+t} x^{1-\beta} \cdot g'(x) \ dx \leqslant \int_{2^k+t}^{n'+t} x^{-\beta} \cdot g(x) \ dx
$$

and therefore

$$
\int_{x=2^k+t}^{n'+t} x^{-\beta} \cdot g(x) \ dx \leqslant \frac{1}{\beta-2} (2^k+t)^{1-\beta} \cdot g(2^k+t)
$$

by plugging it into the integration by parts. Thus, equation (4) yields

$$
\sum_{i=2^k}^{n'} (i+t)^{-\beta} \cdot g(i+t)
$$

$$
\leqslant (2^k+t)^{-\beta} \cdot g(2^k+t) + \frac{1}{\beta-2} (2^k+t)^{1-\beta} \cdot g(2^k+t).
$$

Plugging this into equation (3) now gives

$$
c_1 \cdot n(t+1)^{\beta-1} \sum_{j=k}^{\lfloor \log(n-1) \rfloor} g(2^{j+1}-1) \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta}
$$

$$
\leqslant c \cdot c_1 n(t+1)^{\beta-1} \cdot g(2^k+t) \left((2^k+t)^{-\beta} + \frac{1}{\beta-2} \cdot (2^k+t)^{1-\beta} \right). \qquad (5)
$$

Now we still need a lower bound on $D(k)$. It holds that

$$
D(k) = c_1 \cdot n(t+1)^{\beta-1} \sum_{i=2^k}^{n'} (i+t)^{-\beta}
$$

$$
\geqslant c_1 \cdot n(t+1)^{\beta-1} \frac{1 - \left(\frac{t+2}{t+1} \right)^{1-\beta}}{\beta-1} (2^k+t)^{1-\beta}, \qquad (6)
$$

where the last line follows by observing

$$
\sum_{i=2^k}^{n'} (i+t)^{-\beta} \geqslant \sum_{i=2^k}^{2^{k+1}-1} (i+t)^{-\beta}
$$

$$
\geqslant \int_{2^k}^{2^{k+1}} (i+t)^{-\beta}
$$

$$
= \frac{1}{\beta-1}\left((2^k+t)^{1-\beta} - (2^{k+1}+t)^{1-\beta}\right)
$$

$$
\geqslant \frac{1}{\beta-1}\left(1 - \left(\frac{t+2}{t+1}\right)^{1-\beta}\right)(2^k+t)^{1-\beta}.
$$

This holds because $\frac{2^{k+1}+t}{2^k+t} \geqslant \frac{2+t}{1+t}$, since $2^k \geqslant 1$.

Plugging equation (5) and equation (6) into equation (2) and lower-bounding $\sum_{i=2^k}^{n'} (i+t)^{-\beta}$ with $(2^k+t)^{-\beta}$ gives us an upper bound of

$$
\sum_{x \in S} h(\deg(x))
$$

$$
\leqslant \left(\frac{c \cdot c_1 n(t+1)^{\beta-1} \cdot g(2^k+t)\left((2^k+t)^{-\beta} + \frac{(2^k+t)^{1-\beta}}{\beta-2}\right)}{c_1 \cdot n(t+1)^{\beta-1} \sum_{i=2^k}^{n'} (i+t)^{-\beta}} + C\right) \cdot |S|
$$

$$
\leqslant \left(c\left(1 + \frac{\beta-1}{\beta-2}\frac{1}{1-\left(\frac{t+2}{t+1}\right)^{1-\beta}}\right) \cdot g(2^k+t) + C\right) \cdot |S|
$$

$$
= \left(c \cdot a_{\beta,t} \cdot g(2^k+t) + C\right) \cdot |S| \tag{7}
$$

It now suffices to find an upper bound for 2^k+t, since $g(2^k+t)$ is non-decreasing. Due to $\sum_{x \in S} \deg(x) \geqslant M(n)$ and the choice of k it holds that

$$
c_1 \cdot n(t+1)^{\beta-1} \sum_{j=k-1}^{\lfloor \log(n-1)\rfloor} (2^{j+1}-1) \sum_{i=2^j}^{2^{j+1}-1} (i+t)^{-\beta} \geqslant \sum_{x \in S} \deg(x) \geqslant M(n). \tag{8}
$$

To upper bound the left-hand side, we can use equation (5) with $g(x) = x$ and 2^{k-1} in place of 2^k. It is easy to check, that this function satisfies (i), (ii) with $c = 2$ and (iii) as needed. This yields

$$
M(n) \leqslant 2 \cdot c_1 \frac{\beta-1}{\beta-2} n(t+1)^{\beta-1} \cdot (2^{k-1}+t)^{1-\beta} \cdot (2^{k-1}+t)
$$

$$
\leqslant 2 \cdot c_1 \frac{\beta-1}{\beta-2} n(t+1)^{\beta-1} \cdot 2^{\beta-2} \cdot (2^k+t)^{2-\beta}
$$

or equivalently

$$
(2^k+t) \leqslant \left(c_1 \frac{\beta-1}{\beta-2}\frac{n}{M(n)} \cdot 2^{\beta-1} \cdot (t+1)^{\beta-1}\right)^{\frac{1}{\beta-2}} = b_{c_1,\beta,t} \cdot \left(\frac{n}{2 \cdot M(n)}\right)^{\frac{1}{\beta-2}}. \tag{9}
$$

Now we can plug equation (9) into equation (7) to get the result as desired.

$\square$

5.2 Minimum Dominating Set

As we have stated in Section 5.1, the idea for lower-bounding the size of dominating sets is the same as the one by Shen et al. [46] and by Gast et al. [29]: Every dominating set S has to satisfy $n \leqslant \sum_{x \in S} \deg(x) + 1$. If we assume PLB-U, the Potential Volume Lemma tells us how big S has to be in order to satisfy this inequality.

Theorem 5.3 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the minimum dominating set is of size at least*

$$(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)^{-1} \, n = \Theta(n).$$

Proof Let OPT denote an arbitrary minimum dominating set. It holds that

$$\sum_{x \in \text{OPT}} \deg(x) + 1 \geqslant n$$

and since we assume that there are no nodes of degree 0, it also holds that

$$\sum_{x \in \text{OPT}} \deg(x) \geqslant \frac{n}{2},$$

giving us $M(n) := \frac{n}{2}$. We can choose $h(x) := x + 1$ with $g(x) = x$. Now g satisfies (i), (ii) with $c = 2$ and (iii). With Lemma 5.2 we can now derive

$$n \leqslant \sum_{x \in \text{OPT}} \deg(x) + 1 = \sum_{x \in \text{OPT}} h(\deg(x))$$

$$\leqslant \left(c \cdot a_{\beta,t} \cdot g \left(b_{c_1,\beta,t} \cdot \left(\frac{n}{2 \cdot M(n)} \right)^{\frac{1}{\beta-2}} \right) + C \right) \cdot |\text{OPT}|$$

$$= (2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1) \cdot |\text{OPT}|,$$

which proves the theorem. $\square$

The theorem implies that one is guaranteed a constant approximation factor by simply taking all nodes.

Corollary 5.4 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, every dominating set has an approximation factor of at most*

$$2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1$$

Algorithm 1 Greedy Dominating Set

Require: undirected graph $G = (V, E)$
1:   $C \leftarrow \emptyset$
2:   $D \leftarrow \emptyset$
3:   **while** $|D| < |V|$ **do**
4:      $u \leftarrow \mathrm{argmax}_{v \in (V \setminus C)} \left(|N^+(v) \setminus D|\right)$
5:      $C \leftarrow C \cup \{u\}$
6:      $D \leftarrow D \cup N^+(u)$
7:   **end while**
8:   **return** C

However, we can show that using the classical greedy algorithm (c.f. Algorithm 1) actually guarantees a better approximation factor under PLB-U (see Figure 3). This is the case, because the greedy algorithm is guaranteed to find a solution C with $C \leqslant \sum_{x \in \mathrm{OPT}} H_{\deg(x)+1}$. We can then use the Potential Volume Lemma to upper-bound this expression in terms of $|\mathrm{OPT}|$, resulting in a better approximation factor. The rest of this section is dedicated to showing this result.

The following inequality can be derived from an adaptation of the proof for the greedy SET COVER algorithm to the case of unweighted DOMINATING SET.

Theorem 5.5 ([32]) *Let C be the solution of the greedy algorithm and let OPT be an optimal solution for DOMINATING SET. Then it holds that*

$$|C| \leqslant \sum_{x \in OPT} H_{\deg(x)+1},$$

where H_k is the k-th harmonic number.

Proof The idea of the proof is to distribute the cost of taking a node $v \in C$ amongst the nodes that are newly dominated by v. For example, if the algorithm chooses a node v which newly dominates v, v_1, v_2 and v_3, the four nodes each get a cost of $1/4$. At the end of the algorithm it holds that $\sum_{v \in V} c(v) = |C|$.

Now we look at the optimal solution OPT. Since all nodes $v \in V$ have to be dominated by at least one $x \in \mathrm{OPT}$, we can assign each node to exactly one $x \in \mathrm{OPT}$ in its neighborhood, i.e. we partition the graph into stars $S(x)$ with the nodes x of the optimal solution as their centers.

Now let us analyze how the greedy algorithm constructs its solution. Choose one $x \in \mathrm{OPT}$ arbitrary but fixed. Let us have a look at the time a node $u \in S(x)$ gets dominated. Let $d(x)$ be the number of non-dominated nodes from $S(x)$ right before u gets dominated. Due to the choice of the algorithm, a node v had to be chosen which dominated at least $d(x)$ nodes. This means u gets a cost $c(u)$ of at most $1/d(x)$. Now we look at the nodes from $S(x)$ in reverse order of them getting dominated in the algorithm. The last node to get dominated has a cost of at most 1, the next-to-last node gets a cost of at most $1/2$

and so on. Since $|S(x)| \leqslant \deg(x) + 1$ the costs to cover $S(x)$ are at most $\frac{1}{\deg(x)+1} + \frac{1}{\deg(x)} + \ldots + 1 = H_{\deg(x)+1}$. This gives us the inequality

$$|C| = \sum_{v \in V} c(v) = \sum_{x \in \mathrm{OPT}} \sum_{u \in S(x)} c(u) \leqslant \sum_{x \in \mathrm{OPT}} H_{\deg(x)+1}$$

as desired. $\square$

From the former theorem, one can easily derive the following corollary.

Corollary 5.6 *The greedy algorithm gives a $H_{\Delta+1}$-approximation for* Domi-*nating* Set, *where Δ is the maximum degree of the graph.*

The problem of the corollary is, that it assumes all nodes of an optimal solution to have maximum degree Δ. However, if we assume PLB-U, we can get a better bound on the degrees of nodes in an optimal solution. By using the inequality from Theorem 5.5 together with the Potential Volume Lemma, we can derive the following approximation factor for the greedy algorithm.

Theorem 5.7 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the classical greedy algorithm for* Minimum Dominating Set *(cf. [22]) has an approximation factor of at most*

$$\log_3(5) \cdot a_{\beta,t} \ln (b_{c_1,\beta,t} + 1) + 1 = \Theta(1).$$

Proof From the analysis of the greedy algorithm we know that for its solution C and an optimal solution OPT it holds that

$$|C| \leqslant \sum_{x \in \mathrm{OPT}} H_{\deg(x)+1} \leqslant \sum_{x \in \mathrm{OPT}} \ln(\deg(x) + 1) + 1,$$

where H_k denotes the k-th harmonic number. We can now choose $h(x) = g(x)+1$ with $g(x) = \ln(x + 1)$. $g(x)$ satisfies (i), (ii) with $c = \log_3(5)$ and (iii). As we assume there to be no nodes of degree 0, it holds that

$$\sum_{x \in \mathrm{OPT}} \deg(x) \geqslant \frac{n}{2} =: M(n),$$

since all nodes have to be covered. We can now use Lemma 5.2 with $S = \mathrm{OPT}$ to derive that

$$|C| \leqslant (\log_3(5) \cdot a_{\beta,t} \cdot \ln (b_{c_1,\beta,t} + 1) + 1) |\mathrm{OPT}|. \tag{10}$$

$\square$

It is not easy to see that this is an improvement over the approximation factor implied by the lower bound. Figure 3 displays both approximation factors for several realistic parameter settings. Note that for $\beta \to 2$ both factors approach infinity, although the one of the greedy algorithm does so much slower. For $\beta \to \infty$ the approximation factor from the lower bound approaches

$$8 \cdot (t + 1) + 1$$
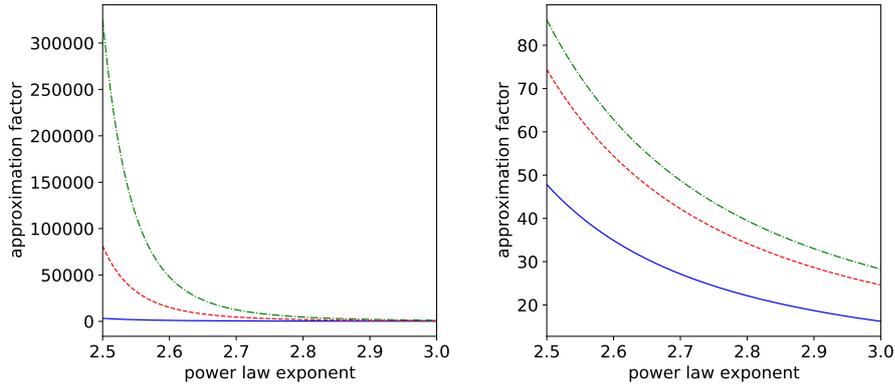
(a) Approximation factor $2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1$ of MDS with PLB-U for $t = 0$ and $\beta \in (2.5, 3)$. We compare $c_1 = 1$ (——), $c_1 = 5$ (– –), and $c_1 = 10$ (–·–).

(b) Approximation factor $\log_3(5) \cdot a_{\beta,t} \cdot \ln\left(b_{c_1,\beta,t} + 1\right) + 1$ of greedy MDS with PLB-U for $t = 0$ and $\beta \in (2.5, 3)$. Again, we compare $c_1 = 1$ (——), $c_1 = 5$ (– –), and $c_1 = 10$ (–·–).

Fig. 3 Comparison of the two approximation factors for Minimum Dominating Set with PLB-U: trivial approximation from Corollary 5.4 (left) and greedy approximation from Theorem 5.7 (right). The parameter sets for both plots are the same.

Graph	Parameters			Lower Bound Approximation	Greedy Approximation
	β	c_1	t		
Youtube	2.34	1.46	1.76	$27.98 \cdot 10^6$	352.27
Epinion	2.08	1.92	3.84	$8.24 \cdot 10^{36}$	11925.88
HepPh	3.22	3.43	24.14	453068.69	422.94
DBLP	3.34	3.61	14.64	75450.87	221.28
CondMat	4.28	4.62	29.12	34285.62	214.18

Table 3 Trivial and Greedy approximation factors for Minimum Dominating Set with PLB-U and parameters from some real-world networks. The fit of those parameters is visualized in Figure 1.

and the approximation factor of the greedy algorithm approaches

$$\log_3(5) \cdot 2 \cdot \ln(2(t+1) + 1) + 1.$$

We also calculated both factors for the fitted PLB-U bounds from Figure 1. See Table 5.2 for a comparison: For all real-world networks the approximation factor guaranteed by the greedy algorithm is orders of magnitude better than the trivial bound.

For Minimum Connected Dominating Set we get a very similar bound. In this variant of Minimum Dominating Set we are looking for a smallest dominating set S whose induced subgraph in G is connected.

Theorem 5.8 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the greedy algorithm for* MINIMUM CONNECTED DOMINATING SET *(cf. [44]) has an approximation factor of at most*

$$2 + \ln\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right) = \Theta(1).$$

Proof From [44, Theorem 3.4] we know that for the solution C of the greedy algorithm and an optimal solution OPT it holds that

$$|C| \leqslant \left(2 + \ln\left(\frac{n}{|\text{OPT}|}\right)\right) |\text{OPT}| \leqslant \left(2 + \ln\left(\frac{\sum_{x \in \text{OPT}} \deg(x) + 1}{|\text{OPT}|}\right)\right) |\text{OPT}|.$$

We can now choose $h(x) = g(x) + 1$ with $g(x) = x$. $g(x)$ satisfies (i), (ii) with $c = 2$ and (iii). As we assume there to be no nodes of degree 0, it holds that

$$\sum_{x \in \text{OPT}} \deg(x) \geqslant \frac{n}{2} =: M(n),$$

since all nodes have to be covered. We can now use Lemma 5.2 with $S = \text{OPT}$ to derive that

$$|C| \leqslant \left(2 + \ln\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)\right) |\text{OPT}|. \tag{11}$$
$\square$

Our main lemma can be used to improve upon a multitude of results. Another example are algorithms which use only local information for optimization problems: Borgs, Brautbar, Chayes, Khanna, and Lucier [10] consider sequential algorithms where the graph topology is unknown and only vertices from the local neighborhood can be added to the solution. We call an algorithm an r^+-*local algorithm* if its local information is limited to $N^r(S)$ plus the degree of the vertices in $N^r(S)$.

First, we analyze the 1^+-local algorithm for MINIMUM DOMINATING SET from [10] (Algorithm 2). The following theorem is from the original paper by Borgs et al. [10].

Theorem 5.9 ([10]) *Let S be the solution of Algorithm 2 and let* OPT *be an optimal solution for* MINIMUM DOMINATING SET. *Then it holds that*

$$\mathbb{E}[|S|] \leqslant 2 \cdot \sum_{x \in OPT} \left(H_{\deg(x)} + 1\right)$$

and that

$$\mathbb{P}\left[|S| > 2 \cdot \sum_{x \in OPT} \left(H_{\deg(x)} + 2\right)\right] \leqslant e^{-|OPT|}$$

Similar to Theorem 5.7 we can show the following result.

Theorem 5.10 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, let S be the solution of Algorithm 2 and let* OPT *be an optimal solution for* DOMINATING SET. *Then it holds that*

$$\mathbb{P}[|S| \geqslant 2\left(2 \cdot a_{\beta,t} \cdot \ln\left(b_{c_1,\beta,t}\right) + 3\right) |OPT|] < e^{-\Omega(n)}.$$

Algorithm 2 1^+-Local Algorithm for (Connected) Dominating Set (1^+-DS)

Require: undirected graph $G = (V, E)$
 1: Initialize $S = \{u\}$, u is an arbitrary node in graph G.
 2: $D \leftarrow \emptyset$
 3: **while** $|D| < |V|$ **do**
 4:     $u \leftarrow \text{argmax}_{v \in (N(S))} \left(|N^+(v) \setminus D| \right)$.
 5:     $S \leftarrow S \cup \{u\}$
 6:     $D \leftarrow D \cup N^+(u)$
 7:     **if** $N(u) \setminus S \neq \emptyset$ **then**
 8:         Choose $v \in N(u) \setminus S$ uniformly at random and add v to S.
 9:     **end if**
10: **end while**
11: **return** S

Proof Due to Theorem 5.9 with probability at least $1 - e^{-|\text{OPT}|}$ it holds that

$$|S| \leqslant 2 \cdot \sum_{x \in \text{OPT}} \left(H_{\deg(x)} + 2 \right) \leqslant 2 \cdot \sum_{x \in \text{OPT}} \left(\ln(\deg(x)) + 3 \right),$$

where H_k denotes the k-th harmonic number. From Theorem 5.3 we know that $|\text{OPT}| = \Theta(n)$, so this probability is $1 - e^{-\Omega(n)}$. Now we choose $h(x) = g(x) + 3$ with $g(x) = \ln(x)$. $g(x)$ satisfies (i), (ii) with $c = 2$ and (iii). We still assume our graph to not have isolated nodes. It therefore holds that

$$\sum_{x \in \text{OPT}} \deg(x) \geqslant \frac{n}{2} =: M(n),$$

because all nodes have to be covered. We use Lemma 5.2 with $S = \text{OPT}$ to get that

$$|S| \leqslant 2 \left(2 \cdot a_{\beta,t} \cdot \ln \left(b_{c_1, \beta, t} \right) + 3 \right) |\text{OPT}|. \tag{12}$$

$\square$

5.3 Maximum Independent Set

In this section we consider MAXIMUM INDEPENDENT SET on networks with PLB-L or PLB-U. We are going to derive lower bounds on the size of maximum and maximal independent sets linear in n as well as a constant approximation factor.

In general graphs the greedy approximation gives us the following result.

Theorem 5.11 ([45]) *The greedy algorithm which prefers smallest node degrees gives a $(\Delta + 1)$-approximation for MIS in graphs of degree at most Δ.*

However, if we assume PLB-L we can achieve lower bounds on the size of a maximum independent set linear in n.

Lemma 5.12 *A graph with the PLB-L property with parameters $\beta > 2$, $c_2 > 0$ and $t \geqslant 0$, has an independent set of size at least $\frac{c_2(t+1)^{\beta-1}}{(t+d_{min})^\beta(d_{min}+1)} \cdot n$ or of size at least $\frac{c_2}{(t+1)} \cdot n$ if we assume G to be connected and $d_{min} = 1$.*

Proof This is easy to see by just counting the number of nodes of degree d_{min}. There are at least

$$c_2 n(t+1)^{\beta-1}(t+d_{min})^{-\beta} = \frac{c_2(t+1)^{\beta-1}}{(t+d_{min})^\beta}n$$

of these nodes. Since each of these nodes can have at most d_{min} other nodes of the same degree as a neighbor, the independent set is at least of size $\frac{c_2(t+1)^{\beta-1}}{(t+d_{min})^\beta(d_{min}+1)}n$. If $d_{min} = 1$ and G is connected, none of the degree-1 nodes can be neighbors, thus giving us $c_2 n(t+1)^{\beta-1}(t+d_{min})^{-\beta} = \frac{c_2}{(t+1)}n$. □

This is also a lower bound on the solution size a greedy algorithm can achieve, if it chooses nodes of minimum (remaining) degree first. We can even go a step further and show that *all* maximal independent sets have to be quite big, even if we only have the PLB-U property. This holds since every maximal independent set is also a dominating set.

Theorem 5.13 *In a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, every maximal independent set is of size at least*

$$\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)^{-1} n = \Theta(n).$$

Proof It holds that every maximal independent set S is also a dominating set. Due to Theorem 5.3, the size of the minimum dominating set is at least

$$\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)^{-1} n = \Theta(n),$$

giving us the result. □

The former theorem implies that every maximal independent set is a constant-factor approximation of the maximum independent set. This also holds for the greedy algorithm.

Corollary 5.14 *In a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, every maximal independent set has an approximation factor of at most*

$$2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1.$$

We can compare these bounds to the ones from Lemma 5.12. For $d_{min} = 1$, it holds that the approximation factor $\frac{(t+d_{min})^\beta(d_{min}+1)}{c_2(t+1)^{\beta-1}}$ simplifies to $\frac{2(t+1)}{c_2}$. This is a realistic parameter as we can see in Figure 1. In Table 5.3 we also compare the two approximation factors from Lemma 5.12 and Theorem 5.13 for the real-world networks we fitted in Figure 1. For all real-world networks the approximation factor from Lemma 5.12 is better. However, the two results have different requirements: Lemma 5.12 requires PLB-L, while Theorem 5.13 requires PLB-U. Thus, they are only comparable for graphs which have both properties.

Graph	Parameters				Lemma 5.12	Theorem 5.13
	β	c_1	c_2	t		
Youtube	2.34	1.46	0.2	1.76	27.6	$27.98 \cdot 10^6$
Epinion	2.08	1.92	0.03	3.84	322.667	$8.24 \cdot 10^{36}$
HepPh	3.22	3.43	0.3	24.14	167.6	453068.69
DBLP	3.34	3.61	0.03	14.64	1042.67	75450.87
CondMat	4.28	4.62	0.27	29.12	223.11	34285.62

Table 4 Approximation factors for MAXIMUM INDEPENDENT SET with PLB-U and PLB-L parameters of some real-world networks. The fit of those parameters is visualized in Figure 1.

5.4 Minimum Vertex Cover

In this section we consider the MINIMUM VERTEX COVER problem on graphs with PLB-U. Again, we can achieve lower bounds on the size of a minimum vertex cover linear in n by noticing that every vertex cover in a graph without isolated vertices is also a dominating set. Although there is a very easy 2-approximation for general graphs by greedily constructing a maximal matching [27], our result implies that in power-law bounded graphs without isolated vertices, *every* vertex cover is a constant-factor approximation of the minimum vertex cover.

It has to be noted that one can use a greedy algorithm similar to Algorithm 1 to achieve the following result.

Theorem 5.15 *The greedy algorithm which prefers highest node degrees gives a H_Δ-approximation for VERTEX COVER, where Δ is the maximum degree of the graph.*

Proof We can analyze the algorithm in a similar manner as in Theorem 5.5. The difference is that now the algorithm considers covered edges instead of dominated vertices. The cost to add a node to the vertex cover is now distributed among its newly covered edges. We can now consider an optimal solution OPT and partition the edges into sets $E(x)$ with the nodes $x \in$ OPT as their common incident nodes. Thus, for a vertex cover C and an optimal solution OPT it holds that

$$|C| = \sum_{e \in E} c(e) = \sum_{x \in \text{OPT}} \sum_{e \in E(x)} c(e) \leqslant \sum_{x \in \text{OPT}} H_{\deg(x)} \leqslant H_\Delta \cdot |\text{OPT}|,$$

Here, we only have $H_{\deg(x)}$ instead of $H_{\deg(x)+1}$, since the costs are only distributed among incident edges and not the inclusive neighborhoods of nodes. This yields the approximation factor as stated. $\qquad \square$

We could now use the Potential Volume Lemma to get an approximation factor similar to the one the greedy algorithm for dominating set achieves on graphs with PLB-U, see Theorem 5.7. It is interesting to see that even this

naive greedy algorithm based on node degrees achieves a constant-factor approximation on graphs with PLB-U, while it only achieves a H_Δ-approximation on general graphs. However, the algorithm does not achieve an approximation factor better than 2. Therefore, we concentrate on lower-bounding the size of *any* vertex cover.

Theorem 5.16 *In a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, the minimum vertex cover is of size at least*

$$\left(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1\right)^{-1} n.$$

Proof The bound follows from Theorem 5.3, since every vertex cover in a graph without isolated vertices is a dominating set. □

Corollary 5.17 *For a graph without loops and isolated vertices and with the PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, every vertex cover has an approximation factor of at most*

$$2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1.$$

6 Hardness of Approximation

In this section we show hardness of approximation for simple graphs with the PLB-(U,L,N) properties. Let us start with a few definitions.

Definition 6.1 (NP optimization problem [19]) An NP optimization problem (NPO) A is a fourtouple $(I, sol, m, goal)$ such that

1. I is the set of instances of A and it is recognizable in polynomial time.
2. Given an instance $x \in I$, $sol(x)$ denotes the set of feasible solutions of x. These solutions are short, that is, a polynomial p exists such that, for any $y \in sol(x)$, $|y| \leqslant p(|x|)$. Moreover, it is decidable in polynomial time whether, for any x and for any y such that $|y| \leqslant p(|x|)$, $y \in sol(x)$.
3. Given an instance x and a feasible solution y of x, $m(x, y)$ denotes the positive integer measure of y. The function m is computable in polynomial time is also called the objective function.
4. $goal \in \{\max, \min\}$.

We can see that MDS, MIS, and MVC are problems in NPO.

Definition 6.2 (Class APX [22]) APX is the class of all NPO problems that have polynomial-time r-approximation for some constant $r > 1$.

The notion of hardness for the approximation class APX is similar to the hardness of the class NP, but instead of employing a polynomial time reduction it uses an approximation-preserving PTAS-reduction.

Definition 6.3 (APX-Completeness [22]) An optimization problem A is said to be APX-complete iff A is in APX and every problem in the class APX is PTAS-reducible to A.

6.1 Approximation Hardness for Simple Graphs

In this section we prove hardness of approximation for the optimization problems we consider. The following definition and lemma by Shen et al [46] provide us with a framework to transfer inapproximability results on classes of graphs to power-law bounded graphs. Note that both the definition and the lemma assume an optimal substructure problem. Due to Shen et al [46] optimal substructure problems are optimization problems whose optimal solution on a graph is the union of all optimal solutions on the maximal connected components of this graph. This includes the three problems we consider. Our idea to use this framework is to "'hide"' graphs from other classes inside power-law bounded graphs. If the optimal solution size of the hidden graph is big enough, the inapproximability of the class it belongs to carries over. The process of hiding a graph inside another one constitutes an *embedded-approximation-preserving reduction* as formalized in the following definition.

Definition 6.4 ([46]) Given an optimal substructure problem O, a reduction from an instance on graph $G = (V, E)$ to another instance on a (power law) graph $G' = (V', E')$ is called *embedded-approximation-preserving* if it satisfies the following properties:

(1) G is a subset of maximal connected components of G';
(2) The optimal solution of O on G', $\mathrm{OPT}(G')$, is upper bounded by $C \cdot \mathrm{OPT}(G)$ where C is a constant correspondent to the growth of the optimal solution.

Having shown an embedded-approximation-preserving reduction, we can use the following lemma to show hardness of approximation.

Lemma 6.5 ([46]) *Given an optimal substructure problem O and a graph G such that no polynomial time algorithm with approximation factor ε exists for O on G unless $P = NP$. Let $\delta = \frac{\varepsilon C}{(C-1)\varepsilon + 1}$ if O is a maximization problem and let $\delta = \frac{\varepsilon + C - 1}{C}$ if O is a minimization problem. If there exists an embedded-approximation-preserving reduction from G to another graph G', where $\mathrm{OPT}(G') \leqslant C \cdot \mathrm{OPT}(G)$, then no polynomial time algorithm with approximation factor δ exists for O on G' unless $P = NP$.*

We will use this framework as follows: First, we show how to reduce cubic graphs to graphs with PLB-U, PLB-L and PLB-N via an embedded-approximation-preserving reduction. Then, we derive the value of C as in Definition 6.4 for each problem we consider. Last, we use Lemma 6.5 together with the known inapproximability results on cubic graphs to derive the approximation hardness on graphs with PLB-U, PLB-L and PLB-N.

We start by showing how to reduce cubic graphs to simple graphs with PLB-U, PLB-L and PLB-N. Then we can use Lemma 6.5 to show inapproximability. The idea of our reduction is as follows: First, we add the cubic graph G. We choose the total number of nodes N of the constructed graph G_{PLB} high enough so that all n nodes of G fit into bucket 1 according to PLB-L and PLB-U. Then, we fill each bucket d until it complies with PLB-L by adding

stars S_{2^d} (star gadgets). If the nodes of degree one we add with star gadgets reach their minimum according to PLB-L, we start adding cliques K_{2^d} (clique gadgets) instead. This is done until all buckets barely comply with PLB-L. Afterward the process is repeated until we reach or exceed N nodes in total for the first time. At this point all buckets comply with both PLB-L and PLB-U. We will show that we have to switch to clique gadgets only for buckets d, where d is a constant. Hence, all nodes in our construction have at most a constant number of neighbors of same or higher degree. Thus, the constructed graph also satisfies PLB-N.

Lemma 6.6 *Any cubic graph G with $|V(G)| = n$ sufficiently large can be reduced to a simple graph G_{PLB} having the PLB-U, PLB-L and PLB-N properties for any $\beta > 2$, any $t \geqslant 0$, any c_3, and any admissable c_1 and c_2. The reduction takes polynomial time and produces a graph with $N = c \cdot n$ nodes, where*

$$c = \left(c_2(t+1)^{\beta-1} \left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta} \right) \right)^{-1}.$$

Proof Suppose we are given β, t, and $c_1 > c_2 > 0$ admissable. Let n be the number of nodes in graph G and let $N \in \mathbb{N}: N = cn$ be the number of nodes in G_{PLB} for some constant c to be determined. Like in Lemma 6.5 we have to ensure a number of conditions to get a valid degree sequence. First, we have to ensure that the bound PLB-U implies on bucket 1 is high enough to encompass all n nodes of the cubic graph G. The only thing we can still choose in order to do so is $N = c \cdot n$. However, if we choose N so that n is very close to the PLB-U bound, it might happen that we actually add more than N nodes. Thus, we choose N so that n is at most as high as the bound implied by PLB-L on bucket 1. Due to the admissability of c_2 this means that the number of nodes we added is smaller than N, or put differently, it ensures $c > 1$. This means that PLB-U is satisfied for that bucket as well, since $c_1 > c_2$. It holds that

$$c_2 N(t+1)^{\beta-1} \sum_{i=2}^{3} (i+t)^{-\beta} = c_2 N(t+1)^{\beta-1} \left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta} \right) \geqslant n.$$

From this we get

$$c \geqslant \left(c_2(t+1)^{\beta-1} \left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta} \right) \right)^{-1}.$$

Moreover, we will choose c equal to that lower bound. Note that for this choice of c it holds that $c \geqslant 1$, since c_2 is supposed to be admissable and thus the PLB-L lower bound for bucket 1 must be smaller than N. Then we choose the maximum degree Δ such that

$$\Delta(G_{PLB}) = N^{\frac{1}{\beta-1}}$$

and $d_{min} = 1$.

Since c_1 and c_2 are admissible, we can fit exactly N nodes between PLB-L and PLB-U. First, we hide G in bucket 1. Then, we try and add for each bucket $d \geqslant 2$ the number of stars of size $2^d + 1$ it needs to reach its lower bound. Since bucket 0 gets all the degree-one nodes of these star gadgets, we have to be careful not to overfill it. Therefore, we start with bucket $\lfloor \log_2(\Delta) \rfloor$ and fill the buckets with star gadgets until bucket 0 has reached its lower bound or we are done. If bucket 0 is full, we switch to clique gadgets: Each bucket $d \geqslant 1$ then gets enough $(2^d + 1)$-cliques to comply to their lower bound. This does not add any more degree-one nodes. In the end, we also fill bucket 0 with enough 2-cliques, if necessary. By filling a bucket d with stars we do not deviate from the (admissable) lower bound of that bucket; by filling it with cliques we might deviate by $2^d - 1$.

Now we want to know when we have to switch to clique gadgets. We will see that this happens at some bucket x, which contains nodes of constant degrees. This means, every node of degree at least 2^{x+1} is center of a star and thus fulfills PLB-N. Nodes of smaller degree satisfy PLB-N as well, since they only have a constant number of neighbors at all.

It holds that we add at most

$$
\sum_{d=x}^{\lfloor \log \Delta \rfloor} 2^d \left\lceil c_2 \cdot N \cdot (t+1)^{\beta-1} \cdot \sum_{i=2^d}^{2^{d+1}-1} (i+t)^{-\beta} \right\rceil
$$
$$
\leqslant \sum_{d=x}^{\lfloor \log \Delta \rfloor} 2^d \left(1 + c_2 \cdot N \cdot (t+1)^{\beta-1} \cdot \sum_{i=2^d}^{2^{d+1}-1} (i+t)^{-\beta} \right)
$$

degree-one nodes if we start at bucket x. Since $2^d \leqslant i$, we get

$$
2 \cdot \Delta + c_2 \cdot N \cdot (t+1)^{\beta-1} \cdot \sum_{i=2^x}^{2^{\lfloor \log \Delta \rfloor + 1}-1} (i+t)^{-\beta+1}
$$

and upper-bounding this sum with an integral yields

$$
2 \cdot \Delta + c_2 \cdot N \cdot (t+1)^{\beta-1} \cdot (2^x + t)^{1-\beta} \left(1 + \frac{2^x + t}{\beta - 2} \right).
$$

This term is at most $\frac{c_2}{t+1} N$ for a sufficiently large constant x and if n is sufficiently large. This holds, since $\Delta = N^{\frac{1}{\beta-1}} = o(N)$. We can choose

$$
x \geqslant \max \left(\log_2(\beta - 2), \frac{1 - \log_2(\beta - 2) + \beta \cdot \log_2(t+1)}{\beta - 2} \right)
$$

for example. This means, we can fill each bucket $d \geqslant x$ to its lower bound with star gadgets. Smaller buckets (with high-degree nodes) can be filled with star gadgets until PLB-L for bucket 0 is reached. Then, those buckets are filled with clique gadgets. This means, we add at most $2^d - 1$ nodes above PLB-L

for those buckets. Since d is constant, the gap between PLB-L and PLB-U for bucket d is linear in N. Thus, for high enough n, the number of nodes in each bucket still complies with both bounds.

It remains to show that the total number of nodes is proportional to $c \cdot n$ for c as chosen before. If we would add a lot more nodes, PLB-L could increase to a point where buckets $d \geqslant x$ do not satisfy it anymore. A similar argument can be made for PLB-U if we add fewer nodes. We know that all buckets are filled to a point just above PLB-L, except for buckets $d \in [2, x]$, which have a total of at most $\sum_{d=2}^{x} 2^d - 1 \leqslant 2^{x+1}$ extra nodes. Since this is constant, the total number of nodes is now roughly the total of all PLB-L bounds. Due to the requirement that c_2 is admissable, N cannot be reached yet after filling all buckets to PLB-L. Thus, we can continue to fill buckets with star gadgets and clique gadgets as before as long as a bucket's upper bounds from PLB-U are not violated. As before, we start with bucket Δ and switch to cliques at bucket $x - 1$, until N is reached or exceeded for the first time. The bucket we switch at stays the same, since the relation between high-degree nodes and degree-1 nodes in PLB-L and PLB-U is the same. This means, if adding stars such that PLB-L is met does not exceed PLB-L for bucket 0, the same holds if we add stars such that PLB-U is met.

We continue adding stars and cliques until N is met or exceeded for the first time. In the worst case, this happens when adding a star for bucket Δ. Therefore, we can exceed N by at most Δ, which does not change a bucket's lower and upper bounds by too much, as long as n is sufficiently large. After adding these extra nodes, we set N to the exact number of nodes we have at that point. We now have

$$
c = \left(c_2(t+1)^{\beta-1} \left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta} \right) \right)^{-1}
$$

from the condition that G has to fit into bucket 1. $\qquad \square$

We are now ready to show inapproximability of MINIMUM DOMINATING SET, MAXIMUM INDEPENDENT SET, and MINIMUM VERTEX COVER on simple graphs with PLB-L, PLB-U, and PLB-N. The inapproximability will carry over from cubic graphs which can be reduced to power-law bounded graphs without adding too many nodes and without increasing the solution size by too much. In the next sections we will derive for each of these problems by which factor C the size of an optimal solution increases. Together with the known lower bounds ε on approximability in cubic graphs, Lemma 6.5 yields a lower bound δ on approximability in power-law bounded graphs.

6.2 Dominating Set

In order to use the reduction framework as described, we need the following inapproximability result.

Theorem 6.7 ([5, 16]) *In graphs with maximum degree 3 it is* NP-*hard to approximate MDS within a factor of $\frac{391}{390}$.*

We can now prove the desired hardness result:

Theorem 6.8 *For every $\beta > 2$ and every $t \geqslant 0$ MINIMUM DOMINATING SET cannot be approximated to within a factor of*

$$1 + \left(390 \cdot \left(\frac{4}{c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)} - 3 \right) \right)^{-1}$$

on simple graphs with PLB-U, PLB-L and PLB-N unless $\mathsf{P} = \mathsf{NP}$.

Proof Lemma 6.6 gives us an embedded-approximation-preserving reduction from a cubic graph G to a simple graph G_{PLB} with the PLB-U, PLB-L and PLB-N properties. Let $\mathrm{OPT}(G)$ and $\mathrm{OPT}(G_{PLB})$ denote the size of a minimum dominating set for G and G_{PLB} respectively. We know that $\mathrm{OPT}(G) \geqslant \frac{n}{4}$, since in a cubic graph each node dominates at most 4 new nodes and all nodes have to be dominated. It now holds that

$$\begin{aligned}
\mathrm{OPT}(G_{PLB}) &\leqslant \mathrm{OPT}(G) + N - n \\
&\leqslant \mathrm{OPT}(G) + (c-1)n \\
&\leqslant \mathrm{OPT}(G) + 4(c-1)\mathrm{OPT}(G) \\
&= (4c-3) \cdot \mathrm{OPT}(G).
\end{aligned}$$

In the context of Definition 6.4 and Lemma 6.5 this means $C = 4c - 3$. Due to Theorem 6.7 it also holds that $\varepsilon = \frac{391}{390}$ in the context of Lemma 6.5. This gives us an approximation hardness of

$$\begin{aligned}
&1 + \frac{\varepsilon - 1}{C} \\
&= 1 + \frac{1}{390 \cdot (4c-3)} \\
&= 1 + \frac{1}{390 \cdot \left(4\left(c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)\right)^{-1} - 3 \right)} \\
&= 1 + \left(390 \cdot \left(\frac{4}{c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)} - 3 \right) \right)^{-1}
\end{aligned}$$

due to our choice of c in Lemma 6.6. $\qquad\square$

6.3 Independent Set

To use the framework of Shen et al. [46] we need the following inapproximability result for MIS.

Theorem 6.9 ([5, 8]) *In 3-bounded graphs it is NP-hard to approximate MIS within a factor of $\frac{140}{139} - \gamma$ for any $\gamma > 0$.*

We can now state the following theorem:

Theorem 6.10 *For every $\beta > 2$ and every $t \geqslant 0$ MAXIMUM INDEPENDENT SET cannot be approximated to within a factor of*

$$1 + \frac{\frac{1}{139} - \gamma}{4\left(\left(c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)\right)^{-1} - 1\right)\left(\frac{140}{139} - \gamma\right) + 1}$$

for any $\gamma > 0$ on simple graphs with PLB-U, PLB-L and PLB-N unless $\mathsf{P} = \mathsf{NP}$.

Proof As before we reduce a cubic graph G to a graph G_{PLB} which fulfills PLB-U, PLB-L and PLB-N using Lemma 6.6. Let $\text{OPT}(G)$ and $\text{OPT}(G_{PLB})$ denote the size of a maximum independent set of G and G_{PLB} respectively. We know that $\text{OPT}(G) \geqslant \frac{n}{4}$. This follows if we greedily add a node to the independent set and then remove it and all its neighbors from the graph. Since every time we do this we remove at most 4 nodes, we get an independent set of size at least $\frac{n}{4}$. Now the following holds

$$\begin{aligned}
\text{OPT}(G_{PLB}) &\leqslant \text{OPT}(G) + N - n \\
&= \text{OPT}(G) + (c-1)\,n \\
&\leqslant \text{OPT}(G) + 4\,(c-1)\,\text{OPT}(G) \\
&= (4c - 3)\,\text{OPT}(G).
\end{aligned}$$

This results in $C = 4c - 3$ in the context of Definition 6.4 and Lemma 6.5. Due to Theorem 6.9 it also holds that $\varepsilon = \frac{140}{139} - \gamma$ for any $\gamma > 0$ in the context of Lemma 6.5. This gives us an approximation hardness of

$$\begin{aligned}
&1 + \frac{\varepsilon - 1}{(C-1)\varepsilon + 1} \\
&= 1 + \frac{\frac{1}{139} - \gamma}{4(c-1)\left(\frac{140}{139} - \gamma\right) + 1} \\
&= 1 + \frac{\frac{1}{139} - \gamma}{4\left(\left(c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)\right)^{-1} - 1\right)\left(\frac{140}{139} - \gamma\right) + 1}
\end{aligned}$$

due to our choice of c in Lemma 6.6.   □

6.4 Vertex Cover

Again, we need the following inapproximability result to use the framework.

Theorem 6.11 ([15]) *In 3-regular graphs MVC is hard to approximate within a factor of $\frac{100}{99}$ unless* $\mathsf{P} = \mathsf{NP}$.

Now we can derive the following result:

Theorem 6.12 *For every $\beta > 2$ and every $t \geqslant 0$ MINIMUM VERTEX COVER cannot be approximated to within a factor of*

$$1 + \left(99 \cdot \left(\frac{2}{c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)} - 1\right)\right)^{-1}$$

on simple graphs with PLB-U, PLB-L and PLB-N unless P = NP.

Proof We use Lemma 6.6 to reduce a cubic graph G to a graph G_{PLB} which fulfills PLB-U, PLB-L and PLB-N. Let $\text{OPT}(G)$ and $\text{OPT}(G_{PLB})$ denote the size of a minimum vertex cover of G and G_{PLB} respectively. We know that $\text{OPT}(G) \geqslant \frac{n}{2}$ for cubic graphs. This follows from a simple counting argument: A cubic graph with n vertices has $m = \frac{3}{2}n$ edges due to the handshake lemma. Since every vertex covers at most 3 edges, one needs at least $\frac{n}{2}$ vertices to cover all edges. It now holds that

$$
\begin{aligned}
\text{OPT}(G_{PLB}) &= \text{OPT}(G) + \text{OPT}(G_{PLB} \backslash G) \\
&\leqslant \text{OPT}(G) + N - n \\
&= \text{OPT}(G) + (c-1)n \\
&\leqslant \text{OPT}(G) + 2(c-1)\text{OPT}(G) \\
&= (2c-1)\text{OPT}(G),
\end{aligned}
$$

which gives us $C = 2c - 1$ in the context of Definition 6.4 and Lemma 6.5. Due to Theorem 6.11 it also holds that $\varepsilon = \frac{100}{99}$ in the context of Lemma 6.5. This gives us an approximation hardness of

$$
\begin{aligned}
1 + \frac{\varepsilon - 1}{C} &= 1 + \frac{1}{99 \cdot C} \\
&= 1 + \frac{1}{99 \cdot (2c-1)} \\
&= 1 + \left(99 \cdot \left(\frac{2}{c_2(t+1)^{\beta-1}\left(\frac{1}{(2+t)^\beta} + \frac{1}{(3+t)^\beta}\right)} - 1\right)\right)^{-1}
\end{aligned}
$$

due to our choice of c in Lemma 6.6.   □

7 Parameterized Complexity

In this section we study the parameterized analogues of MDS, MIS and MVC. We first briefly introduce the concepts concerning parameterized complexity.

Definition 7.1 A *parameterized* problem is a set $P \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet. If $(x, k) \in \Sigma^* \times \mathbb{N}$ is an instance of a parameterized problem, we refer to x as the input and to k as the parameter.

In the framework of parameterized complexity [21], the run-time of an algorithm is viewed as a function of input size and parameter.

Definition 7.2 A parameterized problem $P \subseteq \Sigma^* \times N$ is *fixed-parameter tractable* (FPT) if there is a computable function $f : \mathbb{N} \to \mathbb{N}$, a constant $c \in \mathbb{R}^+$ and an algorithm that, given a pair $(x, k) \in \Sigma^* \times \mathbb{N}$, decides if $(x, k) \in P$ in at most $f(k)n^c$ steps, where n is the length of the input.

The class FPT contains all fixed-parameter tractable problems. There is also a theory of parameterized intractability that allows to identify parameterized problems which do not have *fpt*-algorithms. The parameterized intractable hierarchy is represented by W[i] for $i \geqslant 1$. For an introduction and more details see [21].

7.1 Solution Size as Parameter

One of the well-known parameters for parameterized problems is the solution size. The parameterized version of dominating set is defined as follows:

Problem 7.3 (parameterized DOMINATING SET (p-DOMINATING SET))
Input: A graph G and a parameter $k \in \mathbb{N}$.
Question: Does G have a DOMINATING SET of size at most k?

For general graphs p-DOMINATING SET is known to be $W[2]-$hard [21] i.e, one can not expect an *fpt*-algorithm for this problem. However, for PLB-U graphs there exists a simple *fpt*-algorithm for p-DOMINATING SET. The algorithm works as follows: If the parameter $k < (2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)^{-1} n$, then return NO. Otherwise, check all subsets of size less than k and return YES if one of them is a dominating set. Clearly, the algorithm is correct as it is a brute-force algorithm. It runs in FPT-time, because from Lemma 5.2 we know that the lower-bound on the size of the minimum dominating set is $\Theta(n)$. Thus, the algorithm can return yes only when $k = \Theta(n)$ and clearly it runs for at most $\mathcal{O}(2^n) = 2^{\mathcal{O}(k)}$ time, which is FPT. In the same manner we can get *fpt*-algorithms for MINIMUM VERTEX COVER and MAXIMUM INDEPENDENT SET.

7.2 Parameterization Above the Lower Bound

In the former section we consider the solution size as a parameter and prove that the simple brute-force algorithm is an $fpt-$algorithm. Thus, considering the solution size as a parameter does not seem to be right for PLB-U graphs. Now we consider a different approach for parameterization, known as *parameterization above the lower bound* or *above-guarantee parameterization*. This parameterization was first studied by Mahajan, Raman, and Sikdar [38]. The idea is to have a lower bound k' on the solution size and to use $k - k'$ as the parameter instead of k. Mahajan et al [38] show that an FPT-algorithm in the parameter $k - k'$ exists for all problems in the class MAX SNP.

The complexity class MAX SNP was defined by Papadimitriou and Yannakakis [42] using logical expressiveness. It contains all problems with an L-reduction (linear reduction) to an optimization problem on relational structures that can be expressed in a certain form, for example MAX 3-SAT. Thus, MAX 3-SAT is also MAX SNP-complete. Papadimitriou and Yannakakis [42]

showed that many interesting problems like MAX 3-SAT and MAX CUT are in
MAX SNP. Since we are also using L-reductions for our result, we define them
formally.

Definition 7.4 (L-reduction [38]) Let Q_1 and Q_2 be two optimization
problems. We say Q_1 *L-reduces to* Q_2 if there exist polynomial-time computable
functions f, g and constants $\gamma, \tau > 0$ such that for each instance I_1 of Q_1:

1. $f(I_1) = I_2$ is an instance of Q_2, such that $\mathrm{OPT}(I_2) \leqslant \gamma \cdot \mathrm{OPT}(I_1)$.
2. Given any solution y_2 of I_2, g maps (I_2, y_2) to a solution y_1 of I_1 such that
   $|c_1(I_1, y_1) - \mathrm{OPT}(I_1)| \leqslant \tau \cdot |c_2(I_2, y_2) - \mathrm{OPT}(I_2)|$, where c_1, c_2 are polynomial-
   time computable cost functions.

In order to show that MDS, MVC and MIS have an above-guarantee
parameterization, we use a framework by Mahajan et al [38]. They prove that if
a problem Q is in MAX SNP, then there exists an *fpt*-algorithm for Q above its
MAX 3-SAT lower-bound. MAX K-SAT is the problem of finding an assignment
for a Boolean formula in conjunctive normal form with at most k literals per
clause that satisfies the maximum number of clauses. The MAX 3-SAT lower
bound of a problem in MAX SNP is a lower bound on the optimal solution
size one gets from the L-reduction of that problem to MAX 3-SAT. Such an
L-reduction is always possible, since MAX 3-SAT is MAX SNP-complete. For
MAX 3-SAT with m clauses one can find a solution of size $\left\lfloor \frac{m}{2} \right\rfloor$ in polynomial
time. This lower bound on an optimal solution of the reduced instance implies
a lower bound on an optimal solution of the original instance.

Lemma 7.5 *[38] Let Q be a problem in MAX SNP and (f, g, γ, τ) an $L-$
reduction from Q to MAX 3-SAT. For an instance I of Q, let m_I represent
the MAX 3-SAT-lower bound of I. Then the following problems are in FPT:*

- $L_Q = \{(I, k) : I$ *is an instance of Q and $\mathrm{OPT}(I) \geqslant m_I + k\}$, where Q is a
  maximization problem.*
- $L_Q = \{(I, k) : I$ *is an instance of Q and $\mathrm{OPT}(I) \leqslant m_I + k\}$, where Q is a
  minimization problem.*

Now we are ready to prove the main theorem of this section. We prove that
MDS, MVC and MIS on graphs with PLB-U are in MAX SNP by reducing
them to the MAX SNP-complete problem MAX 2-SAT. This implies that all
three problems we consider have an above-guarantee parameterization due to
Lemma 7.5.

Theorem 7.6 *For a graph without loops and isolated vertices and with the
PLB-U property with parameters $\beta > 2$, $c_1 > 0$ and $t \geqslant 0$, MVC, MIS and
MDS are in MAX SNP.*

Proof To prove this statement for MAXIMUM INDEPENDENT SET we reduce
MIS to MAX 2-SAT using L-reduction. For a given instance I of MIS on a
PLB-U graph $G = (V, E)$ create an instance F of MAX 2-SAT in the following
way:

(a) $\forall v \in V$ create a clause (v)
(b) $\forall (u, v) \in E$ create a clause $(\neg u \vee \neg v)$

Claim There exists an optimal assignment that satisfies all the clauses corresponding to the edges.

Proof (Claim) It is easy to see that a solution which does not satisfy a clause $(\neg u \vee \neg v)$, can be converted to one that does by setting $v = 0$. This makes the clause (v) not satisfied, but satisfies $(\neg u \vee \neg v)$ and potentially other clauses encoding an edge. Therefore, setting the variable $v = 0$ can not decrease the number of satisfied clauses, hence it is a valid transformation. $\qquad\square$

Claim The vertices whose variable is set to 1 form an independent set.

Proof (Claim) We know that, if both vertices of an edge e are set to 1, then the clause corresponding to e will be false, contradicting the former claim. Hence, the vertices whose variables are set to 1 form an independent set. $\quad\square$

It follows that $\text{OPT}(F) = \text{OPT}(I) + |E|$. Let $\bar{d}$ be the average degree of the graph. Theorem 5.13 now yields

$$\text{OPT}(F) = \text{OPT}(I) + |E|$$
$$= \text{OPT}(I)\left(1 + \frac{|E|}{\text{OPT}(I)}\right)$$
$$\leqslant \text{OPT}(I)\left(1 + \frac{\bar{d} \cdot n}{2 \cdot \text{OPT}(I)}\right)$$
$$\leqslant \text{OPT}(I)\left(1 + \frac{\bar{d}}{2}(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)\right)$$

From the above mapping we get $\gamma = (1 + \frac{\bar{d}}{2}(2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1))$.

Now we give a polynomial time algorithm for function g. First, convert the MAX 2-SAT solution to a solution of no smaller size that satisfies all the edge clauses by setting the first variable in each unsatisfied clause to 0. The independent set is the set of nodes whose variables are set to 1. We get

$$|\text{OPT}(I) - c_1(y_1, I)| = |(\text{OPT}(I) + |E|) - (c_1(y_1, I) + |E|)|$$
$$\leqslant |\text{OPT}(F) - c_2(y_2, F)|.$$

Thus, taking $\tau = 1$ satisfies the second condition of an $L - reduction$. This implies that MIS on PLB-U graphs is in MAX SNP.

Similar to the above reduction we can give an $L - reduction$ from MINIMUM VERTEX COVER to MAX 2-SAT. We know that the complement of an independent set is a vertex cover. For a given instance I of MIS on a PLB-U graph $G = (V, E)$ create an instance F of MAX 2-SAT in the following way:

(a) $\forall v \in V$ create a clause $(\bar{v})$
(b) $\forall (u, v) \in E$ create a clause $(u \vee v)$

Now again, as proved in the above reduction, we can prove that there exists an optimal assignment that satisfies all the clauses corresponding to the edges and the variables set to 1 form a vertex cover. It follows that, $\text{OPT}(F) = n - \text{OPT}(I) + |E|$ Thus, from Theorem 5.16 we get,

$$
\begin{aligned}
\text{OPT}(F) &= n - \text{OPT}(I) + |E| \\
&\leqslant \text{OPT}(I)\left(\frac{n + |E|}{\text{OPT}(I)} - 1\right) \\
&\leqslant \text{OPT}(I)\left(\frac{(\frac{\bar{d}}{2} + 1)n}{\text{OPT}(I)} - 1\right) \\
&\leqslant \text{OPT}(I)\left((2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)\left(\frac{\bar{d}}{2} + 1\right) - 1\right).
\end{aligned}
$$

From the above mapping we get $\gamma = (2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1)(\frac{\bar{d}}{2} + 1) - 1$.

Now we give a polynomial time algorithm for function g. First, convert the MAX 2-SAT solution to a solution of no smaller size that satisfies all the edge clauses by setting the first variable in each unsatisfied clause to 1. The vertex cover is the set of nodes whose variables are set to 1. It holds that

$$
\begin{aligned}
|c_1(y_1, I) - \text{OPT}(I)| &= |(n - \text{OPT}(I) + |E|) - (n - c_1(y_1, I) + |E|)| \\
&\leqslant |\text{OPT}(F) - c_2(y_2, F)|.
\end{aligned}
$$

Taking $\tau = 1$ now satisfies the second condition of an $L - reduction$. This implies that MVC on PLB-U graphs is in MAX SNP.

In order to reduce MINIMUM DOMINATING SET to MAX 2-SAT we first reduce it to MAX SAT, then to MAX 3-SAT and finally to MAX 2-SAT.

(a) $\forall v \in V$ create a clause $(\bar{v})$
(b) $\forall v \in V$ create a clause $\left(\bigvee_{u \in N^+(v)} u\right)$

Any solution that does not satisfy a clause encoding the neighborhood of v, can be converted to one that does so by setting v to 1. This makes the clause $(\bar{v})$ become unsatisfied, while at the same time potentially satisfying the clauses encoding other neighborhoods.T Since the number of satisfied clauses can not decrease by this step, it is a valid transformation. After the transformation, the variables set to 1 encode a minimum dominating set.

Now we can substitute each neighborhood clause $(\ell_1 \vee \ell_2 \vee \ell_3 \vee \ldots \vee \ell_k)$ of size $k \geqslant 4$ by the clause group $(\ell_1 \vee \ell_2 \vee y_1) \wedge (\bar{y}_1 \vee \ell_3 \vee y_2) \wedge \ldots \wedge (\bar{y}_{k-2} \vee \ell_{k-1} \vee \ell_k)$. This substitutes each k-clause with $k - 2$ 3-clauses and introduces $k - 2$ new auxiliary variables. Now, if one of those clauses is not satisfied, check if all variables of nodes from the same clause group are set to 0. If so, it is possible to satisfy all of them by setting one of the neighborhood's node variables to 1 and setting the auxiliary variables appropriately. Otherwise, only the auxiliary variables have to be set appropriately to satisfy all clauses from the group. Setting the auxiliary variables appropriately can also be done in polynomial time by going through the clause group and using an

auxiliary variable to satisfy a clause if all other literals in the clause are set to 0. Therefore, this transformation is still valid.

The last reduction step is a reduction from MAX 3-SAT to MAX 2-SAT as done by Garey, Johnson, and Stockmeyer [28]. We substitute each 3-clause $(\ell_1 \vee \ell_2 \vee \ell_3)$ with the following group of 1- and 2-clauses: (z), (ℓ_1), (ℓ_2), (ℓ_3), $(\bar{\ell}_1 \vee \bar{\ell}_2)$, $(\bar{\ell}_1 \vee \bar{\ell}_3)$, $(\bar{\ell}_2 \vee \bar{\ell}_3)$, $(\bar{z} \vee \ell_1)$, $(\bar{z} \vee \ell_2)$, and $(\bar{z} \vee \ell_3)$. If the original clause is satisfied, 7 clauses from the new clause group can be satisfied. Otherwise, only 6 of the new clauses are satisfiable. First, we go through the formula and check, if we can increase the number of satisfied clauses by changing the assignment of one of the newly introduced z-variables. The result should be a formula, where each group has either 6 or 7 satisfied clauses. Then we can check if one clause group belonging to a node neighborhood has only 6 satisfied clauses. This means, that the original clause's literals are all unsatisfied. If this is the case, we can set one of the original clause's node variables v to 1 and fix the y-variables and the z-variable appropriately. By doing so, we can satisfy at least one more clause of the resulting formula, while making only $(\bar{v})$ unsatisfiable. Neighborhoods of sizes 1 and 2 are not transformed and can be checked directly. Therefore, this is still a valid transformation.

Let n_i be the number of nodes of degree i in the original graph G. In the second transformation step, we add $\deg(v) - 1$ many 3-clauses for each node v with $\deg(v) \geqslant 2$. These sum up to $2|E| - |V|$ 3-clauses in total, since we do not have isolated vertices. After the next reduction step, we have $7\,(2|E| - |V|)$ satisfiable 2-clauses from 3-clauses. Furthermore, the n_1 2-clauses for degree-1 nodes and $n - |\mathrm{OPT}(I)|$ 1-clauses encoding the dominating set are satisfiable. This gives us $\mathrm{OPT}(F) = n - \mathrm{OPT}(I) + n_1 + 7\,(2|E| - |V|)$. Let $\bar{d}$ be the average degree of G. With Theorem 5.3 it holds that

$$\mathrm{OPT}(F) = |V| - \mathrm{OPT}(I) + n_1 + 7\,(2|E| - |V|)$$

$$= \mathrm{OPT}(I) \cdot \left(\frac{14|E| - 6n + n_1}{\mathrm{OPT}(I)} - 1 \right)$$

$$\leqslant \mathrm{OPT}(I) \cdot \left(\frac{\left(7\bar{d} - 6 + \frac{c_1}{t+1}\right) n}{\mathrm{OPT}(I)} - 1 \right)$$

$$\leqslant \mathrm{OPT}(I) \cdot \left(\left(7\bar{d} - 6 + \frac{c_1}{t+1}\right) \cdot (2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1) - 1 \right).$$

Now we get $\gamma = \left(7\bar{d} - 6 + \frac{c_1}{t+1}\right) \cdot (2 \cdot a_{\beta,t} \cdot b_{c_1,\beta,t} + 1) - 1$. If we transform a solution of the reduced MAX 2-SAT problem to a dominating set as described, we get the following:

$$|c_1(y_1, I) - \mathrm{OPT}(I)|$$
$$= |(n - \mathrm{OPT}(I) + n_1 + 7\,(2|E| - n)) - (n - c_1(y_1, I) + n_1 + 7\,(2|E| - n))|$$
$$\leqslant |\mathrm{OPT}(F) - c_2(y_2, F)|.$$

This means, we can take $\tau = 1$ to satisfy the conditions of an $L - reduction$ and conclude that MDS on PLB-U graphs is in MAX SNP. $\qquad\square$

8 Discussion and future work

We showed that Chung-Lu Random Graphs, Geometric Inhomogeneous Random Graphs, and Hyperbolic Random Graphs exhibit the PLB-U and PLB-N properties. These results answer an open question posed by Brach et al. [11] positively and imply that their results also hold for the random graphs we considered.

In section 5 we proved that on graphs with the PLB-U property, Minimum Dominating Set, Maximum Independent Set, Minimum Vertex Cover, and some related problems like Connected Dominating Set have optimal solutions of size $\Theta(n)$. This implies that even naive algorithms give a constant-factor approximation on networks with this property.

In hopes of getting better results, i.e. getting a PTAS or making one of the problems polynomial-time-solvable, we introduced the PLB-L property. This property demands a lower bound on the degree distribution which complements the upper bound given by PLB-U. Furthermore, we showed that a range of real networks exhibit both PLB-U and PLB-L, as is evident from Figure 1. However, we were able to prove that the three problems we consider still remain APX-hard, even with all three properties, PLB-U, PLB-N, and PLB-L. This means, these three properties alone do not suffice to get a PTAS for Minimum Dominating Set, Maximum Independent Set, or Minimum Vertex Cover. It would be interesting to see if other deterministic properties that real-world networks exhibit, for example the axiomatic properties from [9], can be used to design more efficient algorithms for the problems we consider.

Finally, we showed that MDS, MIS, and MVC are in MAX SNP on networks with PLB-U. This is the first result which proves that any of these problems is in MAX SNP for a class of power-law graphs. Our results also imply that there is an *fpt*-algorithm which uses solution size above some linear lower bound as the parameter. This algorithm could be considerably faster than a brute force approach. Nevertheless, it remains to derive the exact run time of this *fpt*-algorithm.

References

1. Adamic LA, Buyukkokten O, Adar E (2003) A social network caught in the web. First Monday 8(6)
2. Aiello W, Chung F, Lu L (2000) A random graph model for massive graphs. In: 32nd Symp. Theory of Computing (STOC), pp 171–180
3. Aiello W, Chung F, Lu L (2001) A random graph model for power law graphs. Experimental Mathematics 10(1):53–66
4. Albert R, Barabási AL (2002) Statistical mechanics of complex networks. Reviews of modern physics 74(1):47
5. Alimonti P, Kann V (1997) Hardness of approximating problems on cubic graphs. In: 3rd Italian Conference on Algorithms and Complexity (CIAC), pp 288–298

6. Alon N, Feige U, Wigderson A, Zuckerman D (1995) Derandomized graph products. Computational Complexity 5(1):60–75
7. Barabási AL, Albert R (1999) Emergence of scaling in random networks. Science 286:509–512
8. Berman P, Karpinski M (1999) On some tighter inapproximability results. In: 26th Intl. Coll. Automata, Languages and Programming (ICALP), pp 200–209
9. Borassi M, Crescenzi P, Trevisan L (2017) An axiomatic and an average-case analysis of algorithms and heuristics for metric properties of graphs. In: 28th Symp. Discrete Algorithms (SODA), pp 920–939
10. Borgs C, Brautbar M, Chayes J, Khanna S, Lucier B (2012) The power of local information in social networks, Springer Berlin Heidelberg, pp 406–419
11. Brach P, Cygan M, Lacki J, Sankowski P (2016) Algorithmic complexity of power law networks. In: 27th Symp. Discrete Algorithms (SODA), pp 1306–1325
12. Bringmann K, Keusch R, Lengler J (2015) Geometric inhomogeneous random graphs. CoRR abs/1511.00576
13. Bringmann K, Keusch R, Lengler J (2016) Average distance in a general class of scale-free networks with underlying geometry. CoRR abs/1602.05712
14. Bringmann K, Keusch R, Lengler J (2017) Sampling geometric inhomogeneous random graphs in linear time. In: 25th European Symposium on Algorithms (ESA), vol 87, pp 20:1–20:15
15. Chlebík M, Chlebíková J (2006) Complexity of approximating bounded variants of optimization problems. Theoretical Computer Science 354(3):320–338
16. Chlebík M, Chlebíková J (2008) Approximation hardness of dominating set problems in bounded degree graphs. Information and Computation 206(11):1264–1275
17. Chung F, Lu L (2002) The average distances in random graphs with given expected degrees. Proceedings of the National Academy of Sciences 99(25):15,879–15,882
18. Chung F, Lu L (2002) Connected components in random graphs with given expected degree sequences. Annals of Combinatorics 6(2):125–145
19. Crescenzi P, Kann V (1997) Approximation on the web: A compendium of NP optimization problems. In: Rolim JDP (ed) International Workshop Randomization and Approximation Techniques in Computer Science (RANDOM'97), Lecture Notes in Computer Science, vol 1269, pp 111–118
20. Dinur I, Safra S (2005) On the hardness of approximating minimum vertex cover. Annals of Mathematics 162:439–485
21. Downey RG, Fellows MR (2012) Parameterized complexity. Springer Publishing Company, Incorporated
22. Du DZ, Ko KI, Hu X (2011) Design and analysis of approximation algorithms. Springer Publishing Company, Incorporated

23. Dubhashi D, Panconesi A (2009) Concentration of measure for the analysis of randomized algorithms. Cambridge University Press
24. Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the internet topology. In: Symp. Communications Architectures and Protocols (SIGCOMM), pp 251–262
25. Feige U (1998) A threshold of ln n for approximating set cover. Journal of the ACM 45(4):634–652
26. Ferrante A, Pandurangan G, Park K (2008) On the hardness of optimization in power-law graphs. Theoretical Computer Science 393(1–3):220–230
27. Garey MR, Johnson DS (1990) Computers and Intractability; A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., USA
28. Garey MR, Johnson DS, Stockmeyer LJ (1976) Some simplified NP-complete graph problems. Theoretical Computer Science 1(3):237–267
29. Gast M, Hauptmann M, Karpinski M (2015) Inapproximability of dominating set on power law graphs. Theoretical Computer Science 562(Supplement C):436 – 452
30. Halldórsson MM, Radhakrishnan J (1994) Greed is good: approximating independent sets in sparse and bounded-degree graphs. In: 26th Symp. Theory of Computing (STOC), pp 439–448
31. Hauptmann M, Karpinski M (2015) On the approximability of independent set problem on power law graphs. CoRR abs/1503.02880
32. Kao M (2008) Encyclopedia of algorithms. Springer Publishing Company, Incorporated
33. Koch C, Lengler J (2016) Bootstrap percolation on geometric inhomogeneous random graphs. In: 43rd Intl. Coll. Automata, Languages and Programming (ICALP)
34. Krioukov D, Papadopoulos F, Kitsak M, Vahdat A, Boguná M (2010) Hyperbolic geometry of complex networks. Physical Review E 82(3):036,106
35. Kumar R, Raghavan P, Rajagopalan S, Tomkins A (1999) Trawling the web for emerging cyber-communities. Computer Networks 31(11-16):1481–1493
36. Lenzen C, Wattenhofer R (2010) Minimum dominating set approximation in graphs of bounded arboricity. In: 24th Intl. Symp. Distributed Computing (DISC), pp 510–524
37. Leskovec J, Krevl A (2014) SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data
38. Mahajan M, Raman V, Sikdar S (2009) Parameterizing above or below guaranteed values. Journal of Computer and System Sciences 75(2):137 – 153
39. Newman MEJ (2003) The structure and function of complex networks. SIAM Review 45(2):167–256
40. Newman MEJ (2005) Random graphs as models of networks. Handbook of Graphs and Networks pp 35–68
41. Papadimitriou CH, Steiglitz K (1982) Combinatorial optimization: algorithms and complexity. Prentice-Hall
42. Papadimitriou CH, Yannakakis M (1991) Optimization, approximation, and complexity classes. Journal of Computer and System Sciences 43(3):425–440

43. Phadke AG, Thorp JS (2009) Computer relaying for power systems. John Wiley & Sons, Ltd
44. Ruan L, Du H, Jia X, Wu W, Li Y, Ko KI (2004) A greedy approximation for minimum connected dominating sets. Theoretical Computer Science 329(1–3):325–330
45. Sakai S (2003) A note on greedy algorithms for the maximum weighted independent set problem. Discrete Applied Mathematics 126:313–322
46. Shen Y, Nguyen DT, Xuan Y, Thai MT (2012) New techniques for approximating optimal substructure problems in power-law graphs. Theoretical Computer Science 447:107–119